DM 5,–/SFR. 5,–/öS 40,–

6/84

COMPUTER Praxis

Tips + Tricks aus der internationalen Heimcomputer-Szene

für Atari • Commodore • Sinclair • Texas Instruments

Science-Fiction im Silicon Valley

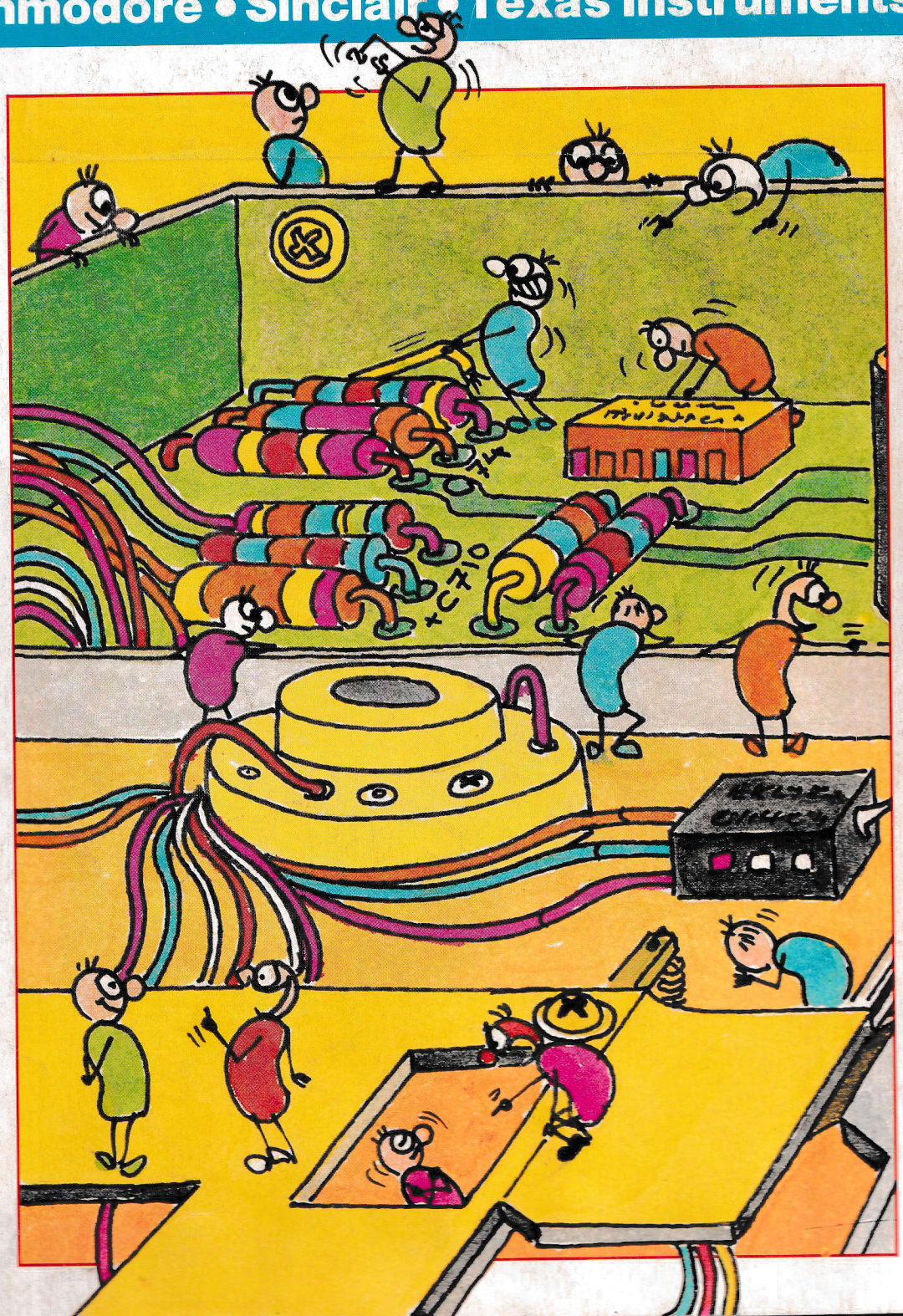
Hier spricht der Spectrum

EPROM statt ROM

Commodores unter sich

Himmelsereignisse am Computer

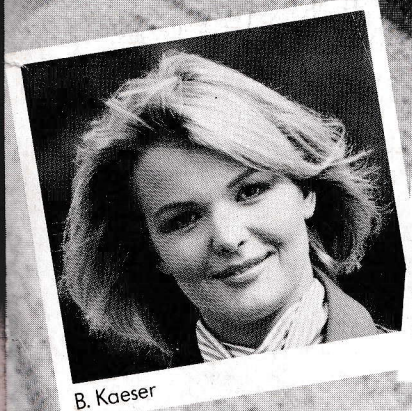
Listings:
Blockade
Fastload
Brick up
Centipede
Eimerkette

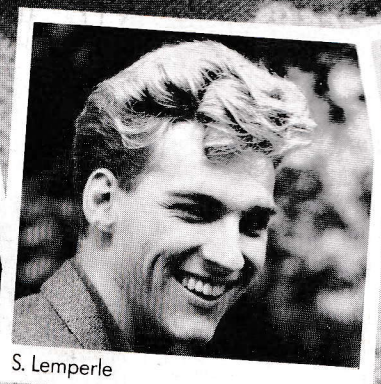


„Na sicher gibt's was gegen Schuppen. Crisan."

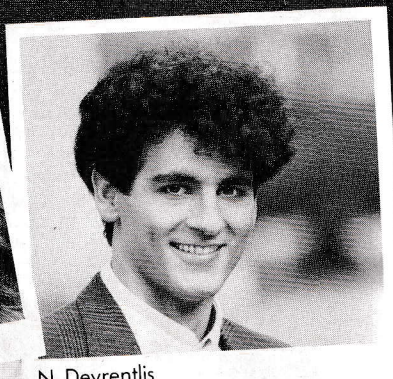
ASCHKE

Crisan, die Kur im Shampoo, löst Haarprobleme gleich beim Waschen.


B. Kaeser


S. Lemperle

M. Oertel


N. Devrentlis

Diese Crisan-Verwender bestätigen es: Crisan löst das Schuppenproblem. Wenn auch Sie mit Crisan zufrieden sind, schreiben Sie uns mit Bild. Für die nächste Anzeige. Wella AG, Postfach 4028, 6100 Darmstadt, Kennwort „Crisan."

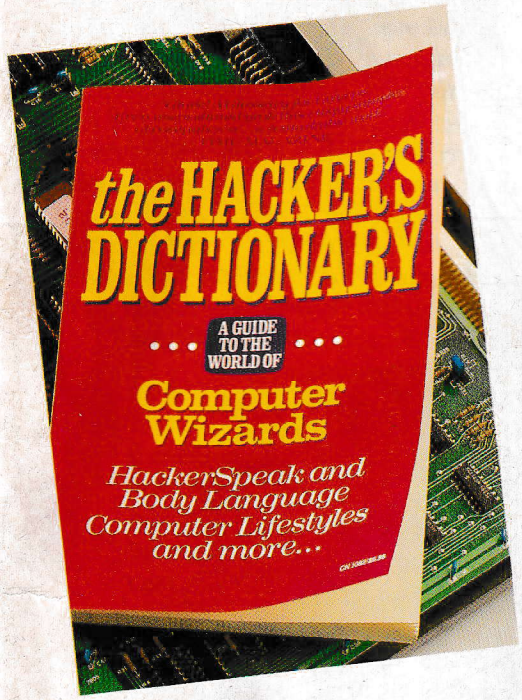
Immer mehr Leute vertrauen Crisan.

WELLA

INHALT

Hallo Computerfreaks 4
Editorial

How to do 4
Tips für unsere Listings



Einen Computer-Zauberer trafen wir in Geoffrey Goodfellow

Hacking durchaus ehrwürdig? 6
Geoffrey Goodfellow steht uns Rede und Antwort

Supercomputer 8
Science-Fiction im Silicon Valley

Kleinanzeigen 40

Leserbriefe 42

Computerclub 44
Gemeinnütziges Computern

BASIC-Kurs 46
Wie sag ich's meinem Computer?

Compyshop 49
Firmenportrait aus dem Westerwald

Schnittstellen unter der Lupe 75
Zweiter Teil

Himmelsereignisse am Homecomputer 78
Das Planetarium im Wohnzimmer

News 81

Vorschau 82

Impressum 82

Atari S.10

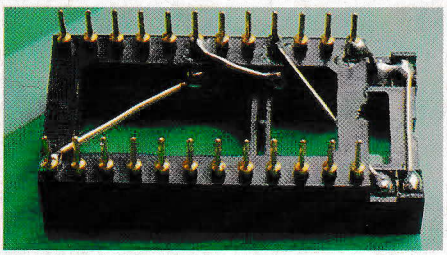
Eine flexible Kiste 10
ATR 8000

Listings 13
Blockade
Fastload

Tips & Tricks 16
Gesangsstunden für euren Computer

Software im Test 22
Jungle Hunt
Squish'em
Ralleyeracer

Buchbesprechung 22
Der Atari Assembler



Zeigt her eure Füßchen: EPROM-Sockelung

Commodore S.26

Hardware 26
EPROM statt ROM

Listings 30
3-D-Labyrinth
Brick up

Tips & Tricks 36
BASIC für Bequeme
Die Commodores unter sich

Software im Test 38
Soccer 2
Castle Wolfenstein
Wacky Waiters

Buchbesprechung 39
BASIC Programme für CBM und VC 20 Computer

Sinclair S.50

Tips & Tricks 50
Die Sprache des Z 80
Scroll auf dem ZX Spectrum
Verständigungsschwierigkeiten

Nicht länger sprachlos: der Spectrum

Hardware 55
Hier spricht der Spectrum

Listings 56
ZX Spectrum Centipede
Drei für einen — den ZX81

Software im Test 62
Jetpac
Gobbleman

Buchbesprechung 62
Das ZX 81 Buch

Texas Instruments S.64

Hardware 64
Woher nehmen und nicht stehlen?

Listings 66
Eimerkette

Software im Test 70
Flugsimulator
Donkey Kong
Defender

Buchbesprechung 70
TI 99/4A

Tips & Tricks 72
to be AND OR NOT to be

HOW TO DO

Hilfreiche Tips zum Umgang mit unseren Listings

Viele von euch haben verwundert nachgefragt, warum unsere Listings so ungewöhnliche Zeichen enthalten und wie man damit umgehen soll.

Das ist eigentlich recht einfach. Wir wollen euch gern gut lesbare Programme liefern, deshalb arbeiten wir mit einem Typenrad-Drucker. Der jedoch 'kennt' einige der Graphik- und Steuerzeichen nicht, deshalb haben wir in solchen Fällen eine 'Ersatzlösung' mit geschweiften Klammern gefunden.

Mit diesen geschweiften Klammern müßt ihr folgendermaßen verfahren: AUF KEINEN FALL einfach abtippen! Steht in den Klammern CHR$( ), nehmt ihr eure ASCII-Tabelle (im Handbuch!) zur Hand und seht nach, was unter der entsprechenden Zahl steht. Dementsprechend übersetzt ihr den Befehl. Nehmen wir an, in einem Atari-Listing steht {CHR$(0)}. In der ASCII-Tabelle findet ihr unter 0 das Herz, also müßt ihr für CHR$(0) ein Herz eingeben.

Oder: ihr findet im Listing einen einfachen Befehl ohne CHR$( ). Den braucht ihr nur auszuführen. Der Befehl {17SPACE}sagt euch, daß ihr 17 mal die Leertaste drücken müßt. Dabei wird gleich ein Vorteil unserer Schreibweise klar: stellt euch vor, wir würden alle 17 Spaces im Heft freilassen. Wer schafft es schon auf Anhieb, die genau auszuzählen??

Wir können uns zwar vorstellen, daß dieses Verfahren für euch ein bißchen ungewohnt ist, trotzdem sind wir ganz zuversichtlich, daß ihr damit zurecht kommt. Und wenn es dann gar nicht klappen will, könnt ihr uns auch gern anrufen, wir versuchen gern, euch bei Problemen zu helfen.

ATARI

Die Cursorsteuerung erscheint in unseren Listings als: LEFT, RIGHT, UP und DOWN.

BUZZER wird durch die Tastenfolge "ESC CTRL 2" eingegeben.

COMMODORE

Die Cursorsteuerung ist wie bei ATARI mit LEFT, RIGHT, UP and DOWN gekennzeichnet.

ARROW UP meint die Taste mit dem Pfeil nach oben (Potenz-Zeichen); die konnte unser Drucker leider nicht darstellen! Die Farbsteuerung entspricht den auf den Tasten befindlichen Kürzeln: BLK, WHT, RED, CYN, PUR, GRN, BLU und YEL.

Das Gleiche gilt für RVS ON und RVS OFF.

Die CHR$-Codes findet ihr im VC20-Handbuch im Anhang J und beim 64er-Handbuch im Anhang F.

WICHTIG!!

Grundsätzlich müssen bei den Commodores alle Programme im Kleinschriftmodus eingegeben werden.

SINCLAIR

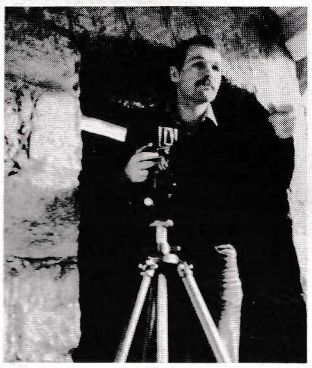
Für die Sinclair-Rechner gelten einige Besonderheiten. So werden die Graphikzeichen nicht durch die entsprechenden CHR$( ) angegeben, sondern durch GRAPHICS und die im Normalmodus zu drückenden Tasten. Handelt es sich um invers darzustellende Zeichen, so ist zwischen GRAPHICS und den Zeichen INVERS eingefügt. Sowohl GRAPHICS als auch INVERS gelten so lange, bis sie durch GRAPHICS OFF bzw. INVERS OFF oder durch das Anführungszeichen aufgehoben werden.

TEXAS INSTRUMENTS

Siehe einleitende Erklärung. ■

Hallo Computerfreaks,

in Heft 4/84 hattet ihr die Gelegenheit, die meisten Mitglieder unseres Redaktionsteams auf dem Editorial-Foto kennenzulernen. Aber da gibt es noch einige andere Leute, die für unsere Zeitschrift wichtig sind. Dazu gehört auch Rolf Seiffe, der auf dem Foto in Heft 4 aus einem ganz einfachen Grund nicht drauf sein kann: er ist unser Photograph und stand hinter der Kamera. Bevor er anfing, für COMPUTER Praxis zu arbeiten, hatte er nie etwas mit Computern zu tun gehabt, aber inzwischen hat auch ihn die Faszination daran gepackt!



Rolf Seiffe in Aktion

Ein großer Teil der Aufnahmen wird natürlich im Studio gemacht, besonders viel Spaß bringt es jedoch, wenn wir den Rolf samt einer Kleinbildkamera einmal in den Wagen packen und zu einem User-Club fahren.

Für unsere Serie über die Computerszene in den Vereinigten Staaten hat Rolf Seiffe auch einen wichtigen Beitrag geliefert: er hat die meisten Fotos mitgebracht, und er hat sogar einige Interviews für uns geführt. Da er selbst längere Zeit in den USA gelebt hat, kennt er dort noch eine Menge Leute, die ihm mit Tips und Ratschlägen zur Seite standen.

Vielleicht lernt ihr Rolf Seiffe ja persönlich kennen, wenn wir wieder einmal mit ihm einen User-Club besuchen.

User-Clubs dürfen sich gerne einmal bei uns melden. Auch telefonisch!
Bis bald, euer

Peter Aldenhoven

Damit Disketten und Computer perfekt harmonieren:

Neu. BASF FlexyDisk® Science.

Die High End Diskette für jedes System.

Jeder Computer stellt seine spezifischen Forderungen an die Disketten. Von Hersteller zu Hersteller sind Laufwerkskonstruktionen und Speicheransteuerung verschieden. Deshalb hat BASF die FlexyDisk Science für Ihren Computer optimal konzipiert. Damit weder Sie noch Ihr Computer auf die High End Diskette mit der extremen Sicherheitsreserve verzichten müssen.

Die BASF-Forschung hat mit der neuen FlexyDisk Science eine spezielle Diskette für den Einsatz in Wissenschaft und Technik entwickelt – getestet auf absolute Datensicherheit und konstantes Langzeitverhalten selbst unter härtesten Einsatzbedingungen. Darüber hinaus führt die intensive Forschungsarbeit der BASF auf dem Gebiet der elektronischen Speichermedien zu einer fortlaufenden Optimierung ihres gesamten Disketten-Programms.

Das neue BASF Disketten-Programm.
Datensicherheit durch Spitzentechnologie.

BASF

Hacking durchaus ehrwürdig?

Computerwissenschaft ist ein exklusives Gebiet. Ist hacking ein erlaubtes Mittel, sich den Zugang zum allgemein verschlossenen Wissen der Zukunft zu verschaffen?

Im Silicon Valley bekam COMPUTER Praxis auch Kontakt zu Geoffrey Goodfellow, dessen Karriere — für diese Ecke der Welt durchaus nichts Ungewöhnliches — mit der Stanfort University zu tun hat. Es ist über zehn Jahre her, er ging damals in die 7. Klasse, da entdeckte er an seiner Schule einen Raum mit Terminals, die mit einer baud-rate von 110 an einen PDP-10 Minicomputer in Stanford angebunden waren. Nach diesem Fund kam er, wie er uns mitteilt, jeden Tag einige Zeit zu früh zur Schule, um über TTY zu spielen. Das ist es, warum man ihn wohl als ersten Hacker der Welt bezeichnen muß.

Vom Hacker zum Spezialisten

Inzwischen hat sich Geoffrey zu einem stattlichen, jungen Mann gemacht. Und er arbeitet seit Jahren offiziell für das Stanford Research Institute (SRI) im Bereich Artificial Intelligence. Doch wie er an diesen Job kam, das ist mehr als ungewöhnlich. In seiner Schule hatte er alles darangesetzt, durch hacking mehr und mehr über die Arbeitsweise des Stanford-Systems herauszubekommen. Auch suchte er die System-Ingenieure Stanfords auf, um die Kenntnisse zu erwerben, die nur ihnen geläufig waren.

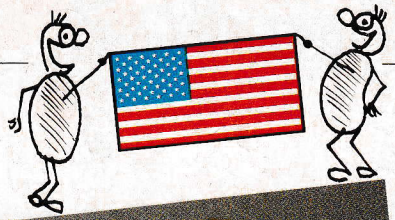
Schritt für Schritt tastete er sich vor, stieg ins ARPANET ein, dem Computer-Network des Verteidigungsministerium, das verschiedenste Computer über die gesamten USA verbindet. Damals waren die Computer-Networks noch nicht so unzugänglich. Es gab viele Möglichkeiten der zumindest geduldeten Fremdbenutzung. Aber Geoff wurde erwischt und plötzlich drohte ihm Gefängnis. Aus Spiel war Ernst geworden! Doch das beschnüffelte System klagte nicht an. Eines Tages wurde Geoff ins SRI bestellt und bekam weitere Ausbildung und einen Job angeboten als einer, der sich im System so gut auskannte wie kaum jemand, eben ein Hacker. Der Teilzeitjob entwickelte sich für ihn so positiv, daß er bald seine Schulausbildung abbrach und aus dem Job einen Beruf machte: Er entwickelt jetzt Sicherheitssysteme für das Department of Defense am SRI. Daß es ihm dabei nicht schlecht geht, zeigt sein teures Hobby — seine Automobilsammlung

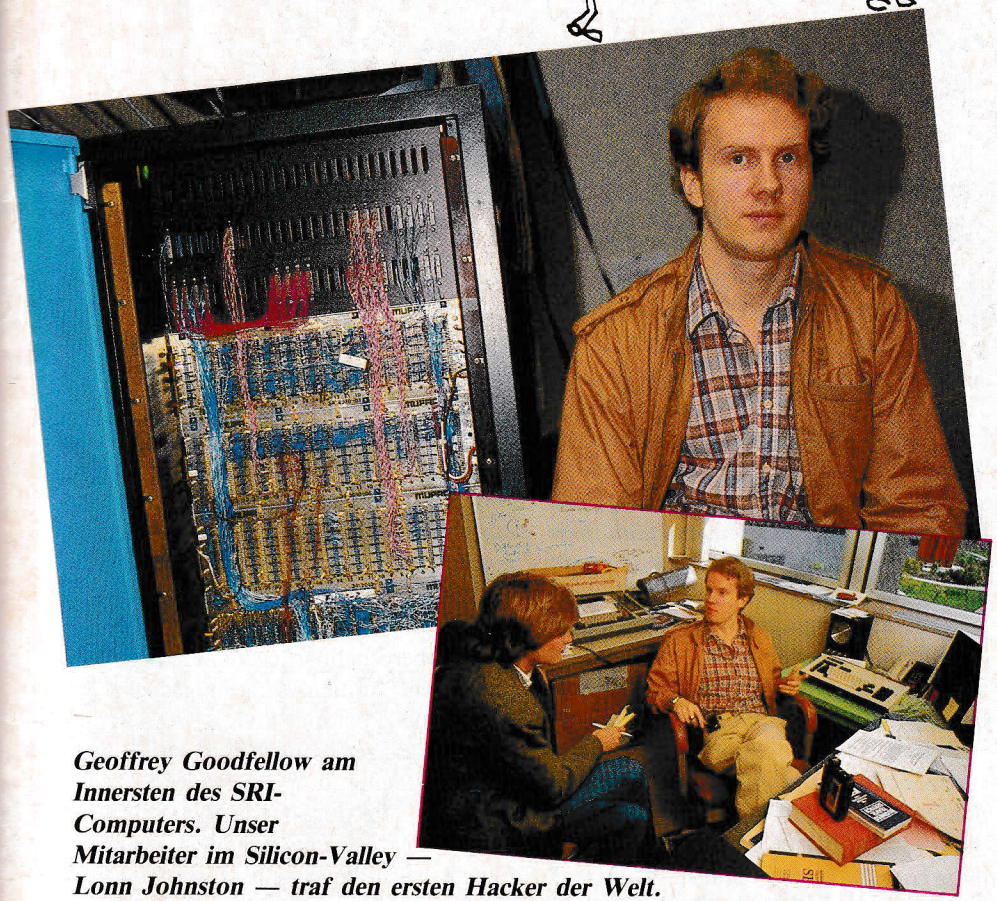
An seine Vergangenheit als Hacker erinnerte er sich lebhaft, als 'unehrenhafte' Hacker mehr und mehr ins Licht der Öffentlichkeit rückten. Die Berichterstattung der Presse dazu hat er in seinem Computer gespeichert, und am Ende unseres Gespräches überreichte COMPUTER Praxis einen fingerdicken Stapel Ausdrucke. Bei dieser allgemeinen Verteufelung der Hacker, die nur

Immer auf Draht, wie es sich für einen gehört, der durch die Hackerschule gegangen ist: Geoff und seine Autotelefone!

von Kriminellen handelte war ihm immer unwohler geworden, und so entschloß er sich, nach Washington D.C. zu fahren, um vor dem Kongreß kundzutun, daß hacking nicht unbedingt von Übel sei.

Er brachte dort seine Befürchtung zum Ausdruck, die Politiker könnten durch die Berichterstattung der Presse





Geoffrey Goodfellow am Innersten des SRI-Computers. Unser Mitarbeiter im Silicon-Valley — Lonn Johnston — traf den ersten Hacker der Welt.

JFCL *(j:-fik':l, jif'k:l) verb.*
To cancel or annul something. "Why don't you jfcl that out?"

The PDP-10 has several instructions that don't do anything (remember that SKIP means "Do not SKIP," as explained in the entry for AOS). However, the fastest do-nothing instruction happens to be JFCL, which stands for "Jump if Flag set and then CLear the flag." This does something useful, but is a very fast no-operation if no flag is specified.

If one wants to patch a program by removing one instruction, the easiest thing to do is to replace the instruction with one that doesn't do anything. Such an instruction is said to have been jfcl'd out. This bit of jargon was then extended metaphorically.

The license plate on hacker Geoff Goodfellow's BMW is JFCL.

dazu veranlaßt sein, übermäßig zu reagieren und damit auch den 'guten' Hackern ein Ende bereiten.

"Die meisten Computersysteme sind totalitären Polizeistaaten vergleichbar, die es lieber sehen, wenn nur die dazu berechtigten Busenfreunde des Systems programmieren dürfen, unentdeckte Genies jedoch außen vor bleiben. Dadurch aber, daß sie den Wert der Hacker nicht erkennen, nutzen sie eine wesentliche Quelle für innovativen Fortschritt überhaupt nicht aus."

Die meisten Hacker, die Geoff kennt, finden die normale Schulausbildung langweilig. Sie lernen am besten, indem sie sich gute Programme gemeinsam mit anderen Hackern erschließen. Aber all das nützt ihnen wenig. Für einen Job müssen sie heute meterlang Diplome vorzeigen können.

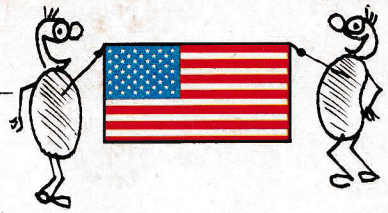
Sein Konzept sieht anders aus. Zum einen sollte die Sicherheit, da wo sie vonnöten erscheint, erhöht werden. Zum Beispiel sollten die Systembenutzer dazu angehalten werden, ernsthafte Paßwörter zu benutzen, und nicht etwa den Namen des Gatten oder ihres Hundes. Versuche mit falschen Paßwörtern sollten dokumentiert werden, nach mehreren Fehlversuchen sollte das System die line unterbrechen. Hacker, die nichts Böses wollten, können veranlaßt werden, ihre Anwesenheit im System deutlich zu machen, wenn sie nichts zu befürchten haben. So kann ein funktionierender Weg aus Vertrauen und Regeln gefunden werden. Ein allgemeines, auf Totalität bedachtes Sicherheitssystem macht jedoch keinen Unterschied zwischen netten Besuchern und argen Einbrechern und fordert Kriminalität geradezu heraus.

Laßt die Hacker in die Systeme!

Überdies gebe es überhaupt kein völliges sicheres Computersystem auf dem Markt. Die Furcht davor, plötzlich ein paar vermaledeite russische Bits im Militärcomputer zu finden, darf nicht dazu führen, ein neurotisches Sicherheitsbedürfnis zu wecken. Statt Polizeistaatmethoden anzuwenden, soll man sich lieber zum Lehrer der Hacker machen und zu ihrem Führer durch die Welt der Computersysteme.

Daß er es ernst meint mit diesem Vorschlag, zeigt nicht nur sein Auftritt vor dem amerikanischen Kongreß, sondern auch das neuerschienene Buch "The Hackers Dictionary" (ISBN 0-06-091082-8), für das er als Co-Autor zeichnet und das er uns stolz überreichte. Ein erster Schritt, aus der Geheimwissenschaft des hackings einen Massensport zu machen: Der Slang der Hacker und obskure Befehle werden erklärt. Auch das sprechende Nummernschild von Geoffs BMW finden wir dort wieder. ■

Lonn Johnston



Supercomputer

Wie Science-Fiction wirkt, was man über die fünfte Computer-Generation hört. An der Stanford University im Silicon Valley wird versucht, diese Zukunftsmusik zu Realität zu machen.

Durch die Flure der Stanford University führt der Weg zu Dr. Harold D. Brown, Professor für Artificial Intelligence, dem Gesprächstpartner von COMPUTER Praxis. Wie in einer deutschen Uni sind die Wände über und über mit Papier behangen. Doch es sind nicht Wandzeitungen mit politischen Parolen, die hier zu Hauf hängen. Nicht die obligate Einladung zur Erstsemester-Fête, "Tippe Examensarbeiten ...", "Folgende Bücher billig von ...". Auf den Zetteln bietet die Computer-Industrie Studierenden Jobs, von denen viele uns zweifeln lassen, ob wir nicht vielleicht doch den falschen Beruf ergriffen haben, so hoch ist der auf ihnen versprochene Verdienst. Nun ja, wir müßten wohl unsere menschlichen Interviewpartner gegen Maschinen eintauschen, aber es wären vielleicht so illustre Gegenüber darunter wie die Computer der 5. Generation, denen Dr. Brown seine wissenschaftlichen Fragen anträgt.

Allwissende Computer?

Die Beschäftigung mit ihnen dürfte nicht uninteressant sein. Obwohl es inzwischen immerhin schon über 70 sogenannte Supercomputer auf der Welt gibt, die in der Lage sind, mehrere 100 Millionen Operationen in der Sekunde durchzuführen, mit 240 000 Chips vollgepackt wie der Cray X-MP, gekühlt mit halogenierten Kohlenwasserstoffen, damit sie vor Denken nicht verrauchen. Ob Stanfords Computer wohl zuverlässig errechnen können, wer 1984 Meister der Fußballbundesliga wird?

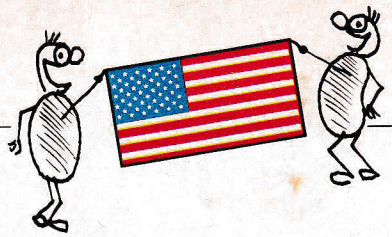
Professor Brown bestätigt unseren Verdacht, daß — wenn überhaupt jemand eine solche Voraussage mit einem hohen Grad an Wahrscheinlichkeit treffen könnte — dies ein Wesen aus Halbleitern sein dürfte. Noch in den 60er Jahren gingen viele Wissenschaftler davon aus, daß man eine riesige Maschine bauen könne, die in der Lage wäre, jede gestellte Aufgabe zu lösen, die man als logisches Problem bezeichnen kann. Allerdings kam man von der Vorstellung eines allwissenden Computers recht bald ab. Schließlich würde der Erwerb von breitgefächertem Wissen einen Computer ebenso einen hohen Zeitaufwand kosten wie einen Menschen. Also kam man bald auf die effektivere Idee, die jungen Supercomputer zu Spezialisten auszubilden. Sie sollten menschliche Aufgaben übernehmen, die kein breites, wohl aber ein tiefgehendes und eng zu begrenzendes Fachwissen benötigen. Professor Browns Aufgabe ist es, für die Supercomputer Software für wissenschaftliche Aufgaben zu entwickeln.

Eines der ersten Programme diese Art war Dendrel. Es ist in der Lage, die Spektralanalyse eines Stoffes durchzuführen und so seine chemische Zusammensetzung zu überprüfen. Es nimmt damit Chemikern zuverlässig einen großen Teil Arbeit ab und gibt ihnen die Möglichkeit, ihre Forschungszeit besser zu nutzen. Geradezu notwendig werden solche Computer natürlich für die Überwachung technischer Abläufe, deren einzelne Schritte in Sekundenbruchteilen aufeinander folgen und deren Fortschreiten Entscheidungen erfordert, die die augenblickliche Beurteilung der jeweils kurzfristig aktuellen Situation voraussetzen. Für die verschiedensten Bereiche kann also Software erstellt werden, — sogenannte expert-systems. Und man könnte sicherlich auch einen Experten für die Fußball-Bundesliga schaffen, wenn jemand das nötige Kleingeld dafür zur Verfügung stellte. Theoretisch könnte so ein System eine ganze Fußballpartie vorausberechnen, die günstigste Taktik entwickeln. So ein System ist ja eben Spezialist darin, eine Fülle von komplexen Zusammenhängen zu einer Berechnung zu verknüpfen. Sicher wird der Ausgang eines Fußballspiels schon vor Anpfiff mitentschieden, die Tore fallen dann aber tatsächlich gerade auf Grund der Unberechenbarkeit eines Spielers, der einen Moment lang ungedeckt ist und seine Position kreativ zum Torschuß umsetzen kann.

Expert-systems

Daher werden solche expert-systems am erfolgversprechendsten dort eingesetzt, wo man meint, das Geschehen mit logischen, das heißt folgerichtigen Denkmodellen vollständig erklären zu können. Die Entwicklung eines Geschehensablaufs muß also komplett durch Bedingungen nach dem Muster "wenn — dann " beschreibbar sein. Expertsystems haben gegenüber dem menschlichen Gehirn als berechnendem Wesen den Vorteil, daß sie ihre Berechnungen stur und solange auch richtig verfolgen, wie das Geschehen den erkannten Gesetzen folgt. In ihrer Berechnungsgeschwindigkeit können sie Echtzeit erreichen, das heißt, den Vorgang ebensoschnell erfassen, wie er abläuft.

Diese Eigenschaft der Supercomputer legt ihr Anwendungsgebiet fest. Die besten Resultate wird man mit ihnen dort erzielen, wo gesetzmäßiger Ablauf eines Geschehens mit seinem technischen Ablauf am weitesten übereinstimmt. Der Ort dafür sind die Labors, wo all die Einflüsse, die außerhalb der zur Berechnung nötigen Formeln liegen, mit Klimaanlagen, gefilterter Luft oder gar Vakuumpumpen abgeschirmt werden können. Unserem Fußballexperten könnte z.B. ein unvorhergesehener Platzregen alle schöne Taktik zunichte machen. Wo



ein Fußballtrainer spontan zur Pause entscheiden kann, andere Stollen unterzuschrauben, ist der Computer ratlos, dessen Wissen nicht breit genug sein kann, jedes unvorhergesehene Ereignis spontan zugunsten der Aufgabenbewältigung zu lösen.

In den Labors ist man jedoch vor plötzlichen Sturzbächen gesichert. Die gute Zusammenarbeit Stanfords mit der Industrie trägt ein Übriges dazu bei, daß die Supercomputer vor allem für die industrielle Technik softwaremäßig ausgebaut werden. Sie liefert auch einen Gutteil des Treibstoffes Geld für die Computerentwicklung. Die beiden Bereiche sind in Stanford so aufs Engste miteinander verknüpft, daß man kaum sagen kann, ob Industrie oder Wissenschaft das Aussehen der Computer von morgen bestimmen.

Auch unsere Frage an Dr. Harold D. Brown klärte das nicht. Warum machte er sich nicht auf und davon von Stanford, wo doch schon seinen Studenten so verheißungsvolle Angebote unterbreitet wurden? Es ist kaum die wissenschaftliche Freiheit, die er als Grund nennt, hier seinen wissenschaftlichen Erkenntnisinteressen nachzugehen. Vielmehr sind es weitaus handfestere Argumente, die er ins Feld führt. In Stanford finde er die nötigen Voraussetzungen für seine Arbeit. Zum einen sind das die vorhandenen Computer, die sich privat längst niemand mehr leisten kann — ein Cray X-MP dürfte über 10 Millionen Dollar kosten —, zum anderen ist das der 'Mittelpunkt des wissenschaftlichen Lebens', den Stanford darstellt. Überhaupt sei es schwer, als Lehrkraft nach Stanford berufen zu werden. Es wird als höchste wissenschaftliche Qualifikation verstanden und steigere durchaus den Marktwert des Forschers. Also ganz ähnlich, wie das bei den Fußballern ist, die in der Nationalelf spielen. Ein zu frühes Fortgehen von Stanford würde bedeuten, daß man seinen Wert abschöpft, bevor er richtigt gewachsen sei.

Und überdies gebe es in Stanford durchaus die Möglichkeit des Nebenerwerbs. Man habe gar die Firma IntelliGenetics gegründet, die Programme für die Molekularbiologie entwickelt, und auch Teknowledge Inc. sei ein Sproß der Artificial-Intelligence-Abteilung. Dort arbeitet auch Professor Brown.

Was die neuen Computer so interessant macht, ist die Tatsache, daß sie im höchsten Maße dazu geeignet sind, dort eingesetzt zu werden, wo sie für mächtige, gesellschaftliche Gruppen größten Nutzen stiften können, nämlich den Feldern Volkswirtschaft und militärische Sicherung. Sie werden von den USA als überlebensnotwendig im 'Wettbewerb zwischen den Staaten' eingestuft, und es sind darum neben den Labors die Gebiete, auf denen man sich stärkstens bemüht, den Zufall auszuschließen. Diesen Zufall, der schon unserem Fußballexperten zum Verhängnis wurde. So finden wir modernste Computer eben gerade auch in Raketenleitsystemen.

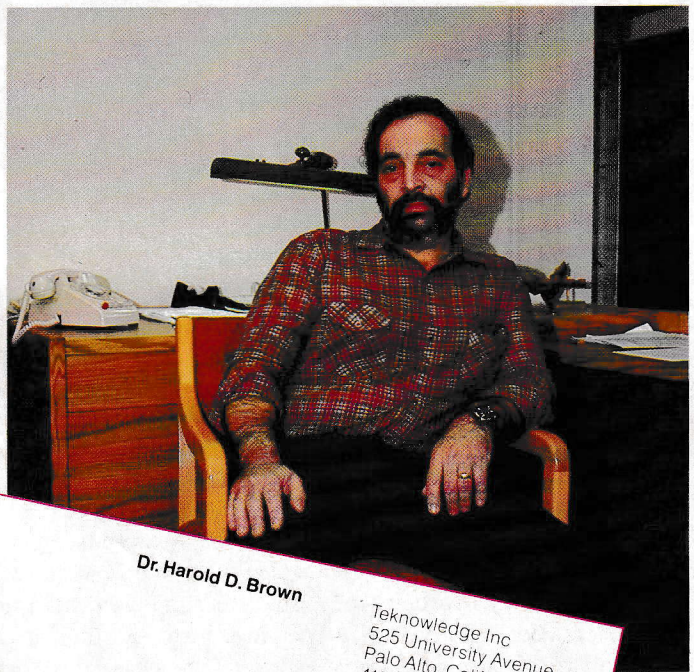
Damit kommt es aber auf dem Gebiet der Computertechnologie zu einem Wettlauf, der durchaus mit dem Rüstungswettlauf vergleichbar ist. Die Phi-

Wettlauf der Maschinen

losophie vom Überleben des Stärkeren findet sich auf wirtschaftlichem Gebiete wieder. Es ist hier vor allem die Angst, mit dem konkurrierenden System Japan im Wettlauf zu unterliegen, das ebenfalls enorme Anstrengungen im Bereich Computertechnologie unternimmt.

Deutschland spiele da in wissenschaftlichen Kreisen kaum eine Rolle, so Professor Brown, der Anfang der 70er Jahre für ein paar Semester an der Technischen Hochschule Aachen war. Damals schon lag die Computerwissenschaft in der BRD etwa fünf Jahre hinter den USA zurück. Im Bereich artificial intelligence dürften es eher ein paar Jährchen mehr gewesen sein ... ■

R.S./H.N.



Dr. Harold D. Brown

Teknowledge Inc
525 University Avenue
Palo Alto, California 94301
415 327-6600

TEKNOWLEDGE

STANFORD UNIV.
COMPUTER SCIENCE
(415) 497-1667

Wissenschaft und Industrie sind gute Nachbarn in Stanfort: In der Person von Professor Brown überschneiden sie sich.

Eine flexible Kiste

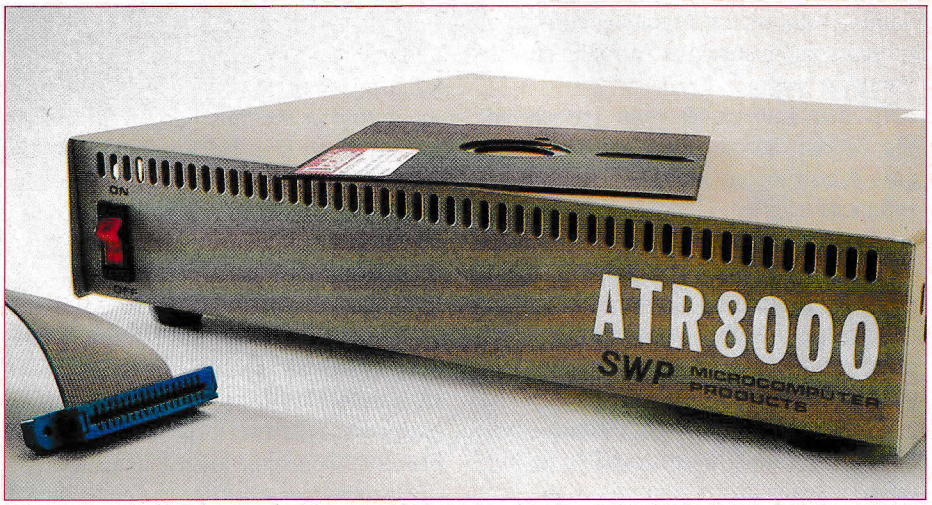
Kann ein Computer auf der Zunge zergehen? Wenn überhaupt, dann dieser!

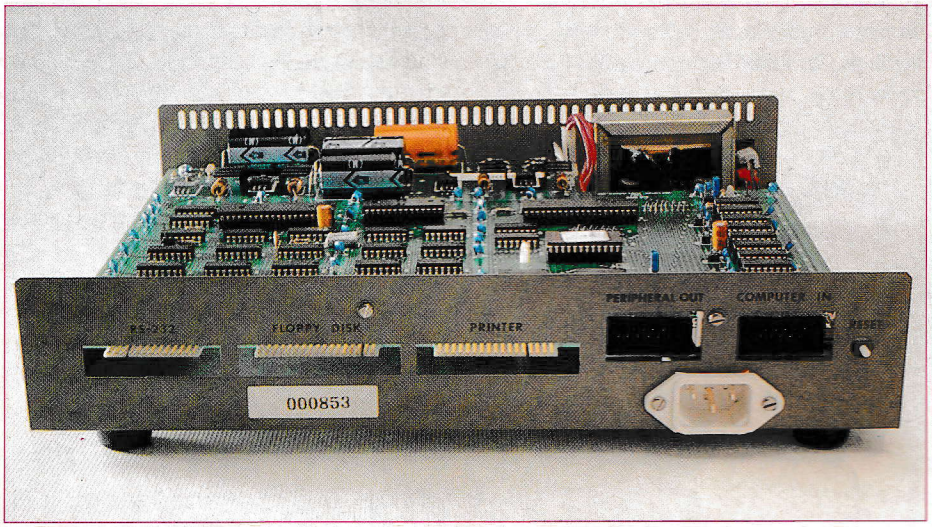
Das ATR8000 ist ein Peripheriegerät der texanischen Firma SWP für den Atari Computer.

Das ATR8000 ist ein Z80 Computer und läuft unter dem Betriebsystem CP/M. Na und?

Das ATR8000 ist ein 8088 Computer und läuft unter den Betriebssysten CP/M-86 und MS-DOS. Wie bitte?

Nun, das ATR ist in jedem Fall ein Z80 Computer mit 16 oder 64 K RAM. Die SWP Co-Prozessor Karte enthält einen 16-Bit Prozessor 8088 und weitere 128 oder 256 K RAM und wird zusätzlich in das ATR eingebaut. Viel mehr ist über das Gerät selber nicht zu sagen. Es ist etwas kleiner als ein Atari 400, sehr stabil und sehr ordentlich aufgebaut und stört den Fernsehempfang in mindestens 15 Meter Umkreis ganz extrem. Was das ATR interessant macht, erfährt man am besten, wenn Schritt für Schritt die Möglichkeiten dieses Geräts demonstriert werden.





Nicht viel größer als der 800 XL, aber leistungsfähig wie ein IBM-PC. Flexibel auf der Anschlußseite durch zwei serielle und drei parallele Ports

Habt ihr schon einmal daran gedacht, einen Drucker an euren Atari anzuschließen, oder einen anderen Computer, ein Modem, einen Sprachsynthesizer? In diesem Fall schreit der Atari nach einer Schnittstelle zur Außenwelt, nach einem Interface, das kompatibel ist zu den Normen Centronics oder RS232. Die einfachste Lösung zu diesem Problem war bisher das 850-Interface von Atari. Leider wird es nicht mehr gebaut. Nehmen wir jetzt einmal an, euch stünde ein ATR8000 in der Grundversion, d.h. Z80 Prozessor und 16 K RAM, zur Verfügung. Man nimmt dann einfach das Kabel aus dem Atari-Systembus und steckt es am ATR in die Buchse "Peripheral Out", der zweite Anschluß wird mit dem mitgelieferten Kabel an den Atari angeschlossen. Alles einschalten und man verfügt jetzt über eine RS232- und eine Centronics-Schnittstelle. Leitet man nun eine Ausgabe an den Drucker, wird man erstaunt bemerken, daß der Atari sich sogleich zurückmeldet und auf weitere Aufgaben wartet, obwohl der Drucker gerade eben mit der ersten Zeile begonnen hat. Das ATR stellt am Druckerport einen mindestens 4 K-Byte großen Druckerpuffer zur Verfügung. Alle Operationen über den seriellen Bus des Atari können, mit Ausnahme des Formatierens einer Diskette, während des Druckens weiterlaufen. In der 64 K-Version speichert der Druckerpuffer sogar ganze 48 K-Byte.

Nun ist ein Drucker beileibe nicht das einzige Gerät, das an einen Computer angeschlossen werden kann. Da wäre z.B. die Diskettenstation. Die 810 oder die neue 1050 von Atari werden an den Systembus angeschlossen, der die Daten seriell mit 19200 Baud überträgt. Um aus der Information auf dem Bus eine Diskettenaufzeichnung zu machen, bedarf es einer Menge Intelligenz, die in diesem Fall von der Diskettenstation aufgebracht werden muß. Und Intelligenz kostet bekanntermaßen Geld; in den Atari-Peripheriegeräten wird das in Form eines eigenen Mikroprozessors und einigen ROMs angelegt. Um diese

Der Telespielversand mit den absoluten Superpreisen

1 Satz Drehregler	45,-
Standard Joystick für Atari	35,-

Sonderaktion für Atari VCS 2600
Action Force von Parker
+ 1 Satz Drehregler nur **79,-**

Kaboom von Activision + 1 Satz Drehregler nur **89,-**

Star Wars Jedi Arena von Parker + 1 Satz Drehregler nur **79,-**

ACTIVISION

für Atari VCS 2600

Free Way	39,-
Boxing	49,-
Skiing	49,-
Tennis	49,-
Ice Hockey	49,-
Grand Prix	49,-
Spider Fighter	49,-
Seaquest	49,-
Oink	49,-
Plaque Attack	49,-
Star Master	49,-
Chopper Command	49,-

Frost Bite	89,-
Decathlon	99,-
H.E.R.O.	99,-
Pitfall II	109,-

für Coleco

Pitfall!	109,-
River Raid	109,-
Beam Rider	109,-

K-tel Double Ender **59,-**

4 verschiedene Spiele für nur **99,-**
— K-tel Double Ender
— Fire Fighter
— Riddle of the Sphinx

IMAGIC

für Mattel

Swords & Serpents	49,-
Safe Cracker	49,-
Beauty & the beast	49,-
Tropic Trouble	49,-
Micro Surgeon	49,-

Neu im Programm:

Software für alle gängigen Computer wie z.B. Commodore VC 20, VC 64, für Atari 600 XL, 800 XL, Texas Instruments TI 99, TI 99/4A, Apple II, Apple IIe etc.

Telefon Service
Immer die aktuellsten Telespiele!
Information
☎ 0511/321414

Bestellschein

Hiermit bestelle ich
mit V-Scheck ☐ (+ DM 4,50 Postgebühren)
als Nachnahme ☐

passend für _____ System

Name Vorname

Straße

PLZ/Ort

Tel.

Einsenden an VIDEOSTAR ★ Prinzenstr. 2
3000 Hannover 1

Bitte kostenlose Telespiel-Preisliste anfordern

Natürlich haben wir auch alle neuen Cassetten im Programm. Rufen Sie uns an!

IMAGIC · INTELLIVISION · Sirius · TIGERVISION · Telesys · PARKER · SPECTRAVISION · DYNAMICS

ACTIVISION · ATARI · CBS COLECO VISION · COMMAVID

Prospekt anfordern!

Prinzenstr. 2 ★ 3000 Hannover 1 ★ ☎ 321414

VIDEOSTAR

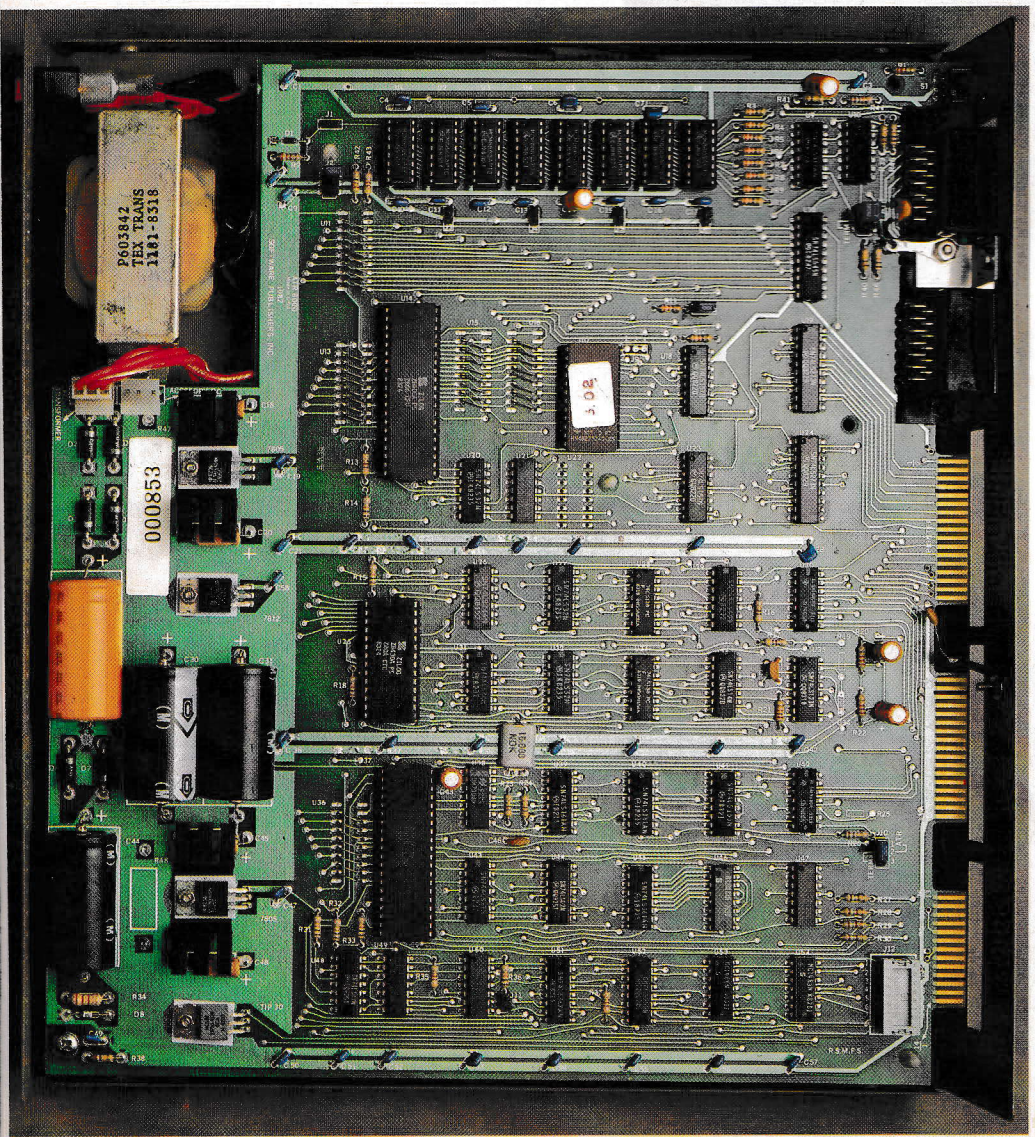
Verschwendung (ein eigener Prozessor für jedes Gerät und zusätzlich ein Floppy-Controller für jedes Laufwerk) zu umgehhen, hat SWP in das ATR einen vom Atari steuerbaren Disk-Controller eingebaut. Auf der Rückseite des Gerätes findet sich ein Standardfloppy-Anschluß, an den vier der Standardlaufwerke angeschlossen werden können.

Auf den Standardlaufwerken lassen sich alle Operationen der Atari-Floppy kompatibel zu dieser ausführen. Wer keinen großen Wert legt auf die Transportierbarkeit seiner Programme zu den bedauernswerten Menschen, die nur eine Atari-Floppy besitzen, der benutzt einfach ein Double-Density Laufwerk mit dem modifizierten Atari DOS und bringt gleich 166 K-Byte auf eine Diskette. Aber es gibt doch da noch die doppelseitigen und die Laufwerke mit 80 Spuren? Hier allerdings enden die Fähigkeiten des Atari DOS, da es nur 720 Sektoren ansprechen kann, wenn auch in Single- oder Double-Density. Man braucht also ein anderes DOS. Verfügbar sind zur Zeit von SWP MYDOS und von OSS das A+ DOS 4.1. Beide erlauben, außer den o.g. 5.25" Laufwerken, sogar die Benutzung von 8" Floppies. Alle Laufwerke, eingeschlossen die von Atari, können beliebig untereinander gemischt angeschlossen werden.

Die Vorteile von Standardlaufwerken gegenüber speziellen Atari-Floppies liegen klar auf der Hand. Von der möglichen größeren Speicherkapazität abgesehen, bieten sie einen deutlichen Preisvorteil und erhöhte Zuverlässigkeit.

Die 16 K-Version des ATR8000 wird mit einem Atari-Verbindungskabel, Netzkabel und Handbuch geliefert. Dieses 85 Seiten umfassende Handbuch, das zur Zeit leider nur in Englisch zu haben ist, enthält eine vollständige Beschreibung der zum Anschluß der verschiedenen Geräte nötigen Schritte. Beispiele unterschiedlicher Ausbaustufen und der nötigen Verbindungen werden gegeben und sogar das Einstellen der Standardlaufwerke findet sich erklärt. Die Pinbelegungen der Anschlüsse, die nötigen Kabelverbindungen und der Ausbau auf 64 K RAM sind erstaunlich gut dokumentiert, ebenso das Ansprechen des ATR über den Bus aus eigenen Programmen und die Monitorroutinen im ROM. Mit der 64 K-Version werden darüber hinaus eine 60 K CP/M Version, viele Utilities (SMODEM37!) und die Digital Research und SWP CP/M Handbücher geliefert.

Wohl strukturiert zeigt sich der innere Aufbau des Erweiterungssystems ATR 8000. Links das großzügig ausgelegte Netzteil, im oberen Drittel das geheimnisvoll abgeklebte EPROM, das steuernde Gehirn des durchaus eigenständigen Gerätes.



ATR 8000 mit Atari 400 als Terminal!

Ein 850 Interfacemodul, ein Druckerpuffer und eine Atari-Diskettenstation kosten zusammen etwa 1600,— DM. Dafür kriegt man mit Sicherheit ein ATR8000. Kauft man zwei Standardlaufwerke anstelle zweier Atari-Floppies, so hat man das Geld für das ATR sogar schon komplett 'erspart'.

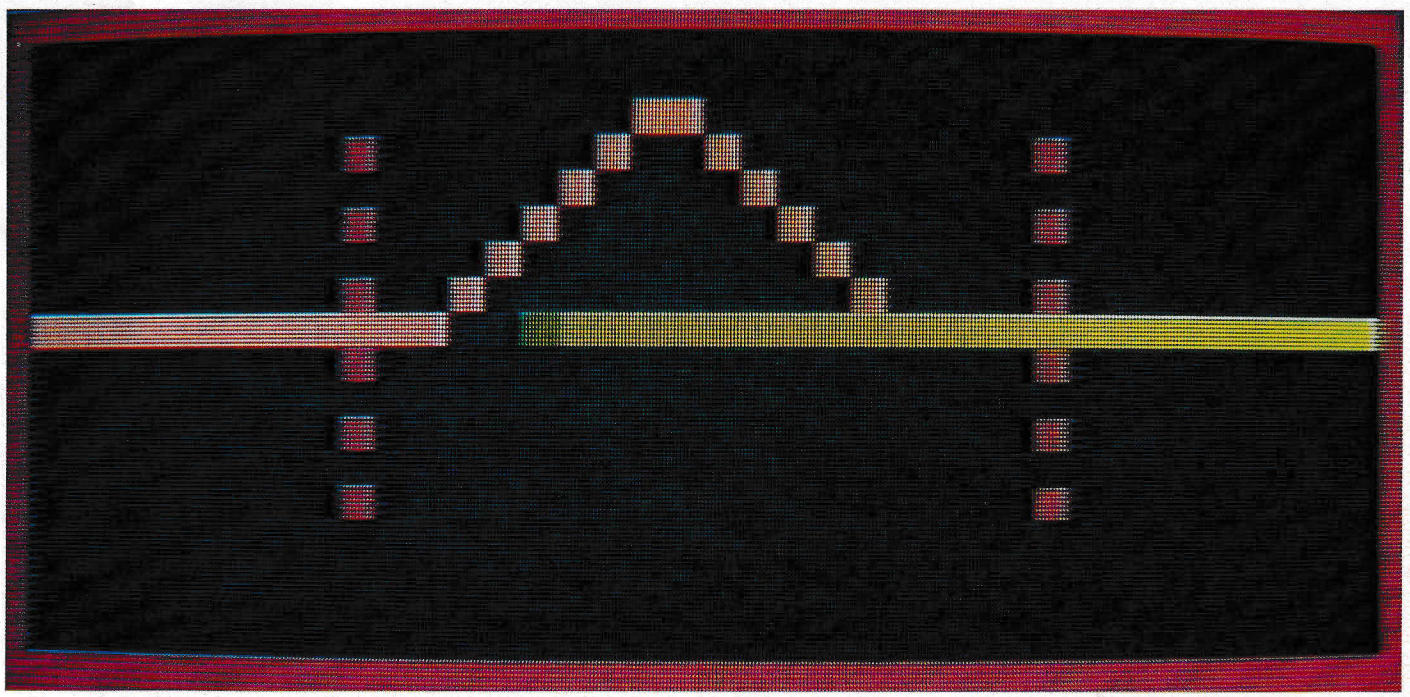
Das ATR ist aber nicht nur ein Peripheriegerät, sondern ein eigenständiger Computer, der den an ihn angeschlossenen "Kleinen", das kann z.B. ein Atari 400 sein, als Terminal benutzt. Es kann ein Atari sein, muß aber nicht. Der VC20 als CP/M Maschine ...? ■

M. A.

Bezugsquelle des ATR8000: Compyshop
Bergstr. 18, 5433 Siershahn, Tel. 02623 / 66 76

BLOCKADE

Ein Basicspiel für zwei Personen



Jeder Spieler baut Stein für Stein eine Mauer, wobei die Ausrichtung der Mauer durch den Joystick gesteuert wird. Die Spielregel für dieses einfache Strategiespiel ist schnell erklärt. Der Mitspieler soll mit der eigenen Mauer blockiert werden, so daß er sich nicht mehr bewegen kann. Jede Berührung mit dem Rand, den Hindernissen, der gegnerischen oder auch der eigenen Figur bedeutet eine verlorene Runde. Das Spiel ist beendet, wenn ein Spieler fünf Runden gewonnen hat.

Nach Eingabe von RUN erscheint auf dem Bildschirm der Titel, bis eine Taste gedrückt wird. Ist eine Runde beendet, wird durch Drücken der Feuertaste die nächste gestartet.

Nachdem das Programm eingegeben ist, solltet ihr es vor dem ersten Start auf Cassette oder Diskette absichern.

Wer gern auf Floppys ausflippt, ist hier richtig:

Computercamp

Ferienzentrum Schloß Dankern

Weitere Informationen über:
Buchungsbüro
Computercamp Ferienzentrum Schloß Dankern
Holztwiete 4 D
2000 Hamburg 52
Tel. (040) 82 79 42

Unser „Programm" überzeugt:

- *das größte Freizeitangebot weit und breit. (Ob Sport, ob Hobby – hier findet jeder, was ihm gefällt.)*
- *kein Hotel, keine Jugendherberge, sondern feriengerechtes Wohnen in Ferienhäusern am See.*
- *spielerische und fachlich qualifizierte Beratung und Betreuung am Computer – und reichlich Zeit zum Programmieren!*
- *attraktive Pauschalangebote für die Ferien.*

Buchungsbüro Computercamp Ferienzentrum Schloß Dankern, Holztwiete 4 D, 2000 Hamburg 52

Antwort-Coupon je früher, desto besser... 5/84

Bitte schicken Sie mir Informationen.

Name

Straße

Alter

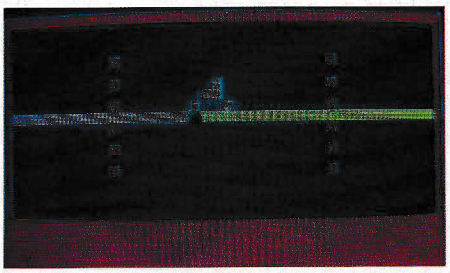
PLZ Ort

besitze Computer Typ

CPr 6/84

Das Programm

Zeile 100 bis 170: Hier wird der Titel angezeigt, wobei die Schleife 160/170 die Farben des Namens ändern.
180 bis 240: Aufbau des Spielfeldes und der Hindernisse.
250 bis 360: Diese Schleife bildet den Hauptteil. Die Joysticks werden abwechselnd gelesen, die Farben der Spieler gesetzt und die Bewegungsrichtung gespeichert. Die Auswertung wird im Subprogramm durchgeführt.
400 bis 440: Hilfsroutine zum Zeichnen der Hindernisse.
460 bis 560: Hier werden die Spielfiguren gezeichnet, der Ton gesteuert und Berührungen erkannt. Die Zeilen 500



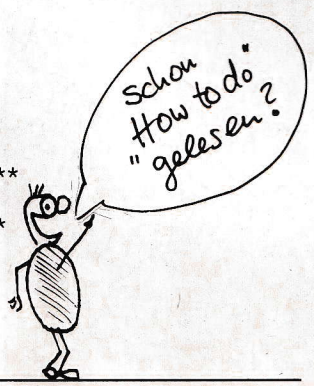
Geschickte Taktik oder Fehler?

und 510 werten die Stellung des Joysticks durch logische Vergleiche aus (hat sich so ganz gut bewährt). In Zeile 550 wird getestet, ob 'etwas im Weg ist'. Bei belegtem Feld wird nach 580 verzweigt, andernfalls Feld belegen und Rückkehr zur Hauptschleife.
580 bis 690: Punktezählung und Anzeige, welcher Spieler einen Fehler machte. Mit POKE 77,0 in Zeile 640 wird das Wechseln der Farben (ATTRAC MODE) verhindert. Hat ein Spieler bereits fünf Punkte, erfolgt ein Programmsprung nach Zeile 710, sonst wartet der Rechner in den Zeilen 680 und 690 auf das Drücken der Feuertaste. An dieser Stelle kann durch Drücken der START-Taste ein neues Spiel begonnen werden.
710 bis 790: Spielende. Das Spielresultat (Punktestand und Gewinner) wird ausgedruckt. Zeile 780 fragt die Feuertaste ab und startet das Programm neu. Durch Einfügen einer Warteschleife (FOR T=1 TO ??:NEXT T) in Zeile 555 kann das Spieltempo gesenkt werden (Zum Üben!).

Aber nun an die Tasten, denn ...

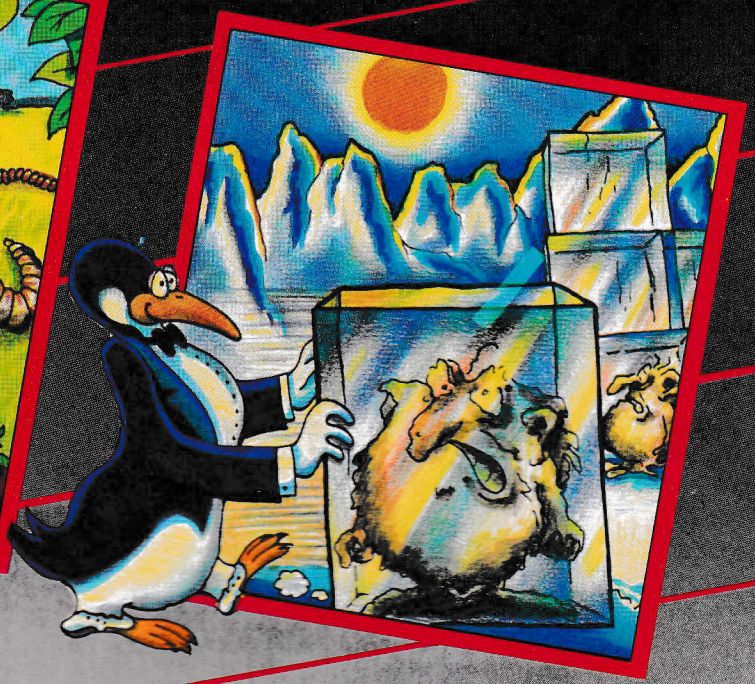
JETZT KOMMT B L O C K A D E

```
100 GRAPHICS 2
110 POKE 752,1:SETCOLOR 0,3,5:SETCOLOR 2,0,0:
    SETCOLOR 3,2,8
120 ? #6;"{4 SPACE}JETZT KOMMT":? " Copyright (C)
    1983 by Gunnar Binder"
130 ? #6:? #6;"{2 SPACE}{INV(b l o c k a d e)}"
140 ? #6;"{20 SPACE}"
150 POKE 764,255
160 T=T+1:SETCOLOR 3,T,5:IF T=15 THEN T=1
170 IF PEEK(764)=255 THEN 160
180 GRAPHICS 3:POKE 752,1
190 SETCOLOR 2,3,5:COLOR 3
200 PLOT 0,19:DRAWTO 0,0:DRAWTO 39,0:DRAWTO 39,19:
    DRAWTO 0,19
210 H1=0:V1=9:H2=39:V2=9
220 B1=7:B2=11
230 H=10:S=0:C=3:GOSUB 400
240 H=29:GOSUB 400
250 REM ***  MAINLOOP  ***
260 S=STICK(0)
270 H=H1:V=V1:B=B1
280 COLOR 1:C=1
290 GOSUB 460
300 H1=H:V1=V:B1=B
310 S=STICK(1)
320 H=H2:V=V2:B=B2
330 COLOR 2:C=2
340 GOSUB 460
350 H2=H:V2=V:B2=B
360 GOTO 260
370 REM ***  SUBPROGRAMME  ***
380 REM
390 REM ***  HINDERNISSE  ***
400 FOR V=4 TO 14 STEP 2
410 COLOR 3
420 GOSUB 460
430 NEXT V
440 RETURN
450 REM ***  TON/BILD  ***
460 IF S=15 THEN S=B
470 IF C=1 THEN SOUND 0,100,14,10
480 IF C=2 THEN SOUND 0,200,14,10
490 IF C=3 THEN SOUND 0,(V*10),10,10
500 H=H+(S=7)+(S=6)+(S=5)-(S=9)-(S=11)-(S=10)
510 V=V+(S=5)+(S=13)+(S=9)-(S=10)-(S=14)-(S=6)
520 IF V<1 OR V>18 THEN 590
530 IF H<1 OR H>38 THEN 590
540 REM ***  TEST AUF CRASH  ***
550 B=S:LOCATE H,V,M:IF M<>0 THEN 580
560 PLOT H,V:RETURN
570 REM ***  CRASH  ***
580 IF C=3 THEN RETURN
590 IF C=1 THEN P2=P2+1
600 IF C=2 THEN P1=P1+1
610 FOR T=10 TO 200:SETCOLOR C-1,6,0:SOUND 0,T,0,10:
    SETCOLOR 0,2,8:SETCOLOR 1,12,10:NEXT T
620 ? "{5 SPACE}PLAYER ";C;" GAME OVER"
630 SOUND 0,0,0,0
640 POKE 77,0:IF P1=5 OR P2=5 THEN 710
650 GRAPHICS 2:SETCOLOR 2,3,5:POKE 752,1:
    ? "{5 SPACE}PLAYER ";C;" GAME OVER"
660 ? #6;"PLAYER 1","PLAYER 2"
670 ? #6:? #6;"{3 SPACE}";P1,"{3 SPACE}";P2
680 IF PEEK(53279)=6 THEN RUN
685 IF STRIG(0)=1 AND STRIG(1)=1 THEN 680
690 POKE 764,255:GOTO 180
700 REM ***  SPIELENDE  ***
710 GRAPHICS 18
720 ? #6;"PLAYER 1","PLAYER 2"
730 IF C=1 THEN C=2:GOTO 750
740 C=1
750 ? #6:? #6;"{3 SPACE}";P1,"{3 SPACE}";P2
760 ? #6;"{20 SPACE}"
770 ? #6;"PLAYER ";C:? #6;"{INV(SIE HABEN GEWONNEN
    !)}"
780 ON (STRIG(0)=0 OR STRIG(1)=0) GOTO 790:GOTO 780
790 RUN
```



G.B.

Action-Software!



JANGLER
Ein Spiel, wo alles durcheinandergeht. Ein Spielfeld voller schlängelnder Raupen, die unter Kontrolle gebracht werden müssen, um den High Score zu erzielen. Ein Spiel, das eine neue Dimension für den Spielfreak setzt.
ZX Spectrum 48k
Art.-Nr. SP232 DM **25,–**

PINGO
Lenken Sie Ihren Pinguin so durch das Treibeis, daß Sie den Sno-bees entgehen. Punkte sammeln Sie aber nur, wenn Sno-bees ins Wasser geworfen oder eingefroren werden. Ein kalter Spaß.
ZX Spectrum 48k
Art.-Nr. SP231 DM **25,–**

MUSIC COMPOSER
Komponieren mit allen Noten und sämtlichen Vorzeichen über 5 Oktaven; Anzeige von 3 parallelen Notensystemen auf dem Bildschirm mit dreistimmigem Abspielen der Musikstücke ... das ist Music Composer!
ZX Spectrum 48k
Art.-Nr. SP240 DM **35,–**

SPACE MISSILE COMMAND
Verteidigen Sie sechs Städte gegen den zerstörenden Meteoritenansturm aus fernen Galaxien. Schätzen Sie die Flugbahnen der Planetoiden und peilen Sie den Zielpunkt mit Ihrem lenkbaren Fadenkreuz, das die Richtung Ihres Lasers bestimmt, an. Hervorragende, farbige Pixelgraphik!
ZX Spectrum 48k
Art.-Nr. SP233 DM **25,–**

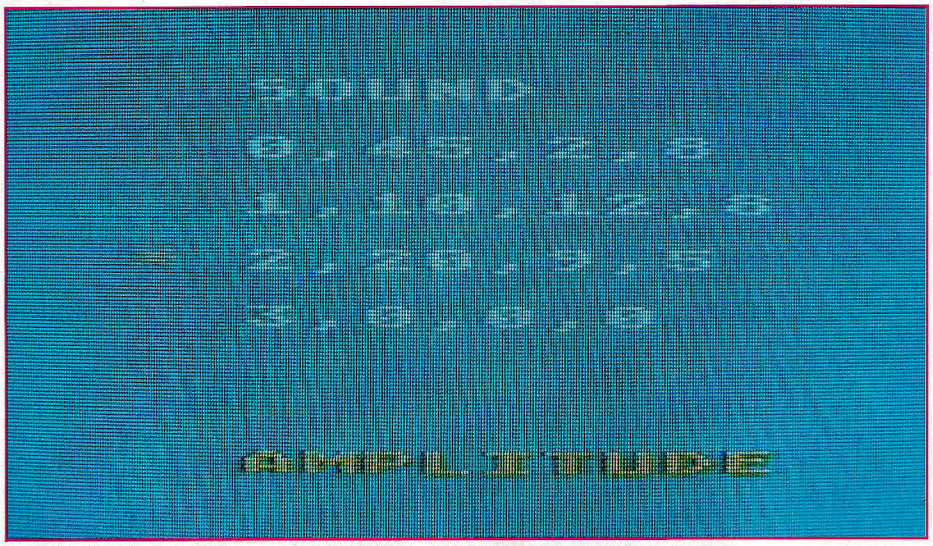
Erhältlich im Fachhandel oder direkt bei uns:

profisoft

Sutthauser Str. 50-52, 4500 Osnabrück
Tel. (0541) 53905, Telex 94966 profis d

Gesangstunden für euren Computer

Konzert für Atari und Joystick



Sich aus der Vielzahl der Möglichkeiten einen bestimmten Sound zu mixen, kann zu einer langwierigen Suche ausarten. Mit 'Sound-Check' jedoch könnt ihr rumprobieren, die Veränderungen sofort hören und, auch nicht zu verachten, die eingestellten SOUND-Parameter direkt ablesen. Das Programm läuft auf allen Atari. Es setzt allerdings einen Joystick in der ersten Buchse voraus.

So läuft's

Nach LOAD und RUN sagt 'Sound-Check' Guten Tag. Spätestens jetzt muß der Joystick angeschlossen werden. Wenn ihr ihn betätigt, kommt ihr ins Auswahlbild. Joystick nach rechts oder links gedrückt: Die verschiedenen Steuermöglichkeiten werden angezeigt. SOUND 0—3 für die einzelne Ansteuerung, ALLE zum gemeinsamen Regeln aller vier Generatoren auf einmal, und zu TAKT kommen wir später.

Wollt ihr die gerade angezeigte Auswahl treffen, drückt auf den Feuerkopf eures Joystick. Wenn ihr ihn wieder losläßt, seht ihr das Steuerbild. Darauf werden die momentanen Werte aller Generatoren angezeigt, und zwar in der Reihenfolge, wie sie im SOUND-Statement anzugeben sind: Generatornummer (0—4), Frequenz (0—255), Verzerrung (0—14), Amplitude (0—15). Der angesteuerte Generator ist mit einem blinkenden Stern gekennzeichnet, falls ihr nicht gerade alle gemeinsam fahrt.

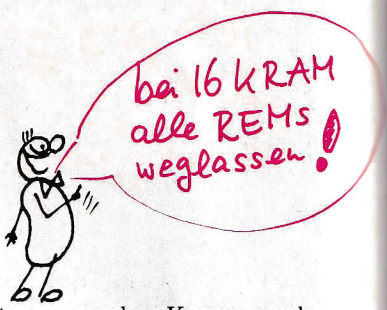
Am unteren Bildrand ist aufgeführt, welcher der drei Werte Frequenz, Verzerrung, Amplitude momentan geregelt werden kann. Joystick nach rechts oder links bedeutet: Auswahl eines anderen Wertes, den ihr dann (Joystick nach vorn: erhöhen, nach hinten: verringern) ändern könnt. Die Frequenzänderung könnt ihr dadurch beschleunigen, daß ihr den Knopf drückt, wenn der Stick vorn oder hinten ist.

Zurück ins Auswahlbild: Joystick in Ruhestellung, Knopf drücken.

Nun noch zur Auswahl TAKT: Hier gelten die Werte Frequenz und Verzerrung, die ihr vorher eingestellt habt. Allerdings verläuft die Lautstärke in Form einer Kurve, deren Länge und Richtung ihr einstellen könnt (Stick nach links: ansteigend, nach rechts: fallend, nach vorn: Verlängerung der Kurve, nach hinten: Verkürzung). Zurück ins Auswahlbild: Joystick in Ruhestellung und Knopf drücken.

Ihr werdet sehen, so schwierig, wie es im Moment aussieht, ist das gar nicht.

Wenn ihr mit dem Rumhorchen fertig seid: ins Auswahlbild und Joystick nach hinten.



So wurde programmiert

Um eine möglichst übersichtliche Struktur zu erhalten, haben wir im Programm folgende Blöcke gebildet: Hauptprogramm (Zeilen 1170—1400), Abstimmen der Generatoren (1500—2120), Taktsteuerung (2200—2710), Titelbild (2800—3110), Endebild (3200—3290). Im eigentlichen Hauptprogramm findet sich neben dem Aufruf der einzelnen Unterprogramme nur das Auswahlbild mit der Auswahl der Generatoren (0—3, alle, Takt) über die Variable mit dem Namen POSO.

Auf folgende Besonderheit möchten wir hinweisen: wir haben den Feuerknopf "entprellt" (z.B. in Zeilen 1360 und 1510 bis 1520). Wenn er gedrückt wird, erscheint, sozusagen als Bestätigung, daß etwas passiert, bereits die neue Hintergrundfarbe; aber weiter geht es erst, wenn er wieder losgelassen wird. Dieser Kniff soll dafür sorgen, daß man beliebig lange auf den Knopf drücken kann, ohne daß der 'Sound-Check' ins Auswahlbild und sofort wieder hinaus springt. Auch haben wir die Weiterschaltung der Wahlmöglichkeiten mit einer Verzögerungsschleife und einer "Entprellung" versehen, damit ihr einigermaßen in Ruhe einstellen könnt.

Um noch diesen oder jenen Hinweis ins Listing zu bringen, haben wir eine ganze Menge REM's reingeschrieben. Falls ihr zu faul seid, sie mit abzutippen, laßt sie ruhig aus, ebenso bei nur 16K RAM Speicherplatz. Es ist Ehrensache, daß wir keine REM-Zeile mit GOSUB oder GOTO anspringen.

Viel Spaß mit dem Listing und — die Ohren gespitzt!!

SOUND-CHECK

```
1000 REM ***     SOUNDCHECK       ***
1010 REM * VARIABLE:
1020 REM * - AIMP AMPLITUDE TAKT
1030 REM * - DIRE TAKT STEIGEND ODER FALLEND
1040 REM * - DISP LAUFVARIABLE ZUR TITELGESTALTUNG
1050 REM * - IMPS IMPULSDAUER
1060 REM * - INCR ERHOEHUNGSFAKTOR
1070 REM * - POSO ZEIGER FUER GENERATOREN (0-3) ODER
1080 REM *        ALLE GENERATOREN(4) ODER TAKT (5)
1090 REM * - SOPT ZEIGER FUER OPTIONSWAHL
1100 REM * - STAB ZWEIDMENSIONALE TABELLE
1110 REM *        (X,0)=FREQUENZ GENERATOR X
1120 REM *        (X,1)=VERZERRUNG GENERATOR X
1130 REM *        (X,2)=AMPLITUDE GENERATOR X
1140 REM DIMENSIONIEREN WERTETABELLE FUER GENERATOREN
1150 DIM STAB(3,2)
1160 GOSUB 2810: REM * TITEL
1170 REM * HAUPTPROGRAMM
1180 GRAPHICS 2+16
1190 SETCOLOR 0,8,14:SETCOLOR 1,1,6:SETCOLOR 4,4,1
1200 IF STICK(0)=7 THEN POSO=POSO+1
1210 IF STICK(0)=11 THEN POSO=POSO-1
1220 IF STICK(0)=13 THEN 3210: REM STICK ZURUECK=ENDE
1230 IF POSO>5 THEN POSO=0
1240 IF POSO<0 THEN POSO=5
1250 IF POSO<4 THEN 1310
1260 IF POSO>4 THEN 1290
1270 POSITION 15,1:PRINT #6,"---------{13 SPACE}
     ALLE{14 SPACE}---------"
1280 GOTO 1320
1290 POSITION 15,1:PRINT #6,"---------{13 SPACE}
     takt{14 SPACE}---------"
1300 GOTO 1320
1310 POSITION 15,1:PRINT #6,"---------{12 SPACE}
     SOUND ";POSO;"{12 SPACE}---------"
1320 POSITION 1,6:PRINT #6,"{14 SPACE}AUSWAHL:"
1330 POSITION 1,8:PRINT #6,"{14 SPACE}< STICK >
     {31 SPACE}KNOPF{2 SPACE}->"
1340 IF STRIG(0)=1 THEN 1380
1350 REM * BEI AUSWAHL TAKT SPEZIELLES UNTERPROGRAMM
1360 IF POSO>4 THEN GOSUB 2210
1370 IF POSO<5 THEN GOSUB 1510
1380 FOR I1=1 TO 80: NEXT I1:´REM * VERZOEGERUNG
1390 GOTO 1190
1400 REM * ENDE HAUPTPROGRAMM
1500 REM * UNTERPROGRAMM ABSTIMMEN GENERATOR
1510 GRAPHICS 1+16: SETCOLOR 0,0,0: SETCOLOR 1,4,9:
     SETCOLOR 2,4,9: SETCOLOR 4,0,4
1520 IF STRIG(0)=0 THEN 1520:
     REM * WEITERGEHEN ERST NACH LOESEN DES KNOPFES
1530 REM * SOPT
1540 REM    0=FREQUENZ
1550 REM * 1=VERZERRUNG
1560 REM * 2=AMPLITUDE (LAUTSTAERKE)
1570 SOPT=0
1580 INCR=1
1590 REM * VERLASSEN: KNOPF DRUECKEN BEI STICK IN RUHE
1600 IF STRIG(0)=0 AND STICK(0)=15 THEN 2090
1610 REM * BESCHLEUNIGUNG
1620 IF STRIG(0)=0 AND SOPT=0 THEN INCR=10
1630 IF STICK(0)<>14 AND STICK(0)<>13 THEN INCR=0
1640 REM * WENN STICK ZURUECK, DANN ERHOEHUNG NEGATIV
1650 IF STICK(0)=13 THEN INCR=-INCR
1660 REM * OPTION : STICK SEITLICH
1670 IF STICK(0)=7 THEN SOPT=SOPT+1
1680 IF STICK(0)=11 THEN SOPT=SOPT-1
1690 IF STICK(0)=7 OR STICK(0)=11 THEN 1690
1700 IF SOPT>2 THEN SOPT=0
1710 IF SOPT<0 THEN SOPT=2
1720 REM * VERAENDERN DER WERTE
1730 REM * BEI 4 ALLE GENERATOREN
1740 IF POSO>3 THEN 1760
1750 I1=POS: GOTO 1770
1760 FOR I1=0 TO 3
1770 STAB(I1,SOPT)=STAB(I1,SOPT)+INCR
1780 IF STAB(I1,SOPT)<0 THEN STAB(I1,SOPT)=0
1790 ON SOPT GOTO 1820,1840
1800 IF STAB(I1,SOPT)>255 THEN STAB(I1,SOPT)=255
1810 GOTO 1850
1820 IF STAB(I1,SOPT)>14 THEN STAB(I1,SOPT)=14
1830 GOTO 1850
1840 IF STAB(I1,SOPT)>15 THEN STAB(I1,SOPT)=15
1850 IF POSO>3 THEN NEXT I1
1860 POSITION 16,7: PRINT #6,"SOUND{21 SPACE}";
1870 PRINT #6,"{4 SPACE}0,";STAB(0,0);",";STAB(0,1);
     ",";STAB(0,2);"{7 SPACE}";;
```


HAASE-Computersysteme – Ihr ATARI-Fachmann:

CPM für Atari
Rana-Floppy
Touch-Tablett
Erweiterungsber.

dt. Literatur

viele Adventure und
Actiongarnes

Basic XL
Compiler

Telefonmodem
Drucker

Flightsimulator, Dimension X, Moon Patrol, Dark Crystal, N. Y. C.

und das alles bei **HAASE-Computersysteme,** Wiedfeldstr. 11, 4300 Essen 1
Info anfordern: Telefon 02 01 – 42 25 75

```
1880 POSITION 16,11:PRINT #6,"1,";STAB(1,0);","
     ;STAB(1,1);",";STAB(1,2);"{7 SPACE}";
1890 POSITION 16,13:PRINT #6,"2,";STAB(2,0);","
     ;STAB(2,1);",";STAB(2,2);"{7 SPACE}";
1900 POSITION 16,15:PRINT #6,"3,";STAB(3,0);",";
     STAB(3,1);",";STAB(3,2);"{7 SPACE}";
1910 IF STRIG(0)=0 THEN 2030
1920 REM * MARKIEREN DES GENERATORS
1930 IF POS0=4 THEN 1950
1940 POSITION 14,(POS0*2)+9: PRINT #6,"{INV(*)}"
1950 FOR I1=1 TO 20: NEXT I1: REM * VERZOEGERUNG
1960 POSITION 14,(POS0*2)+9: PRINT #6," "
1970 REM * ANZEIGE AKTUELLE OPTION
1980 ON SOPT GOTO 2000,2010
1990 POSITION 16,20: PRINT #6,"frequenz{2 SPACE}":
     GOTO 2030
2000 POSITION 16,20: PRINT #6,"verzerrung":
     GOTO 2030
2010 POSITION 16,20: PRINT #6,"amplitude "
2020 REM * ANSTEUERN GENERATOREN MIT AKTUELLEN WERTEN
2030 FOR I1=0 TO 3
2040 SOUND I1,STAB(I1,0),STAB(I1,1),STAB(I1,2)
2050 NEXT I1
2060 REM * SCHLEIFE BILDEN
2070 GOTO 1580
2080 REM * VERLASSEN UNTERPROGRAMM
2090 GRAPHICS 2+16
2100 SETCOLOR 4,4,1
2110 IF STRIG(0)=0 THEN 2110
2120 RETURN : REM * ENDE UNTERPROGRAMM
2200 REM * UNTERPROGRAMM TAKT
2210 GRAPHICS 1+16
2220 SETCOLOR 0,0,0: SETCOLOR 1,4,9:
     SETCOLOR 2,4,9: SETCOLOR 4,0,4
2230 IF STRIG(0)=0 THEN 2230:
     REM * WEITERGEHEN ERST NACH LOESEN DES KNOPFES
2240 REM * IMPS DAUER DES IMPULSES
2250 REM * DIRE 0=FALLEND
2260 REM *      1=STEIGEND
2270 INCR=1
2280 REM * VERLASSEN: KNOPF DRUECKEN BEI STICK IN RUHE
2290 IF STRIG(0)=0 AND STICK(0)=15 THEN 2680
2300 IF STICK(0)<>14 AND STICK(0)<>13 THEN INCR=0
2310 REM * WENN STICK ZURUECK, DANN ERHOEHUNG NEGATIV
2320 IF STICK(0)=13 THEN INCR=-INCR
2330 REM * VERAENDERN DER WERTE
2340 IMPS=IMPS+INCR
2350 IF IMPS<1 THEN IMPS=1
2360 IF IMPS>20 THEN IMPS=20
2370 REM * FORM: STICK SEITLICH
2380 IF STICK(0)=7 THEN DIRE=0
2390 IF STICK(0)=11 THEN DIRE=1
2400 POSITION 16,7: PRINT #6,"SOUND{21 SPACE}";
2410 PRINT #6,"{4 SPACE}0,";STAB(0,0);",";STAB(0,1);
     ",";STAB(0,2);"{7 SPACE}";;
2420 POSITION 16,11:PRINT #6,"1,";STAB(1,0);",";
     STAB(1,1);",";STAB(1,2);"{7 SPACE}";
2430 POSITION 16,13:PRINT #6,"2,";STAB(2,0);",";
     STAB(2,1);",";STAB(2,2);"{7 SPACE}";
2440 POSITION 16,15:PRINT #6,"3,";STAB(3,0);",";
     STAB(3,1);",";STAB(3,2);"{7 SPACE}";
2450 REM * ANZEIGE RICHTUNG UND DAUER
2460 IF DIRE=1 THEN 2480
2470 POSITION 16,20:PRINT #6,"fallend{3 SPACE}";IMPS;
     "{3 SPACE}":GOTO 2490
2480 POSITION 16,20:PRINT #6,"steigend{2 SPACE}";IMPS;
     "{3 SPACE}"
2490 INCR=15/IMPS: IF IMPS=1 THEN INCR=15/2
2500 AIMP=14
2510 IF DIRE=1 THEN INCR=-INCR: AIMP=0
2520 REM * KLANGSCHLEIFE
2530 FOR I1=1 TO IMPS
2540 IF STICK(0)<>15 OR STRIG(0)=0 THEN 2620
2550 AIMP=AIMP-INCR
2560 IF AIMP<0 THEN AIMP=0
2570 IF AIMP>15 THEN AIMP=15
2580 FOR I2=0 TO 3
2590 IF STAB(I2,2)>0 THEN
     SOUND I2,STAB(I2,0),STAB(I2,1),AIMP
2600 NEXT I2
2610 NEXT I1
2620 FOR I1=0 TO 3
2630 SOUND I1,0,0,0
2640 NEXT I1
2650 FOR I1=1 TO IMPS*5: NEXT I1
2660 INCR=0: GOTO 2270
2670 REM * VERLASSEN UNTERPROGRAMM
2680 GRAPHICS 2+16
2690 SETCOLOR 4,4,1
2700 IF STRIG(0)=0 THEN 2700
2710 RETURN : REM * ENDE UNTERPROGRAMM
2800 REM * UNTERPROGRAMM TITEL
2810 STAB(0,0)=250: STAB(0,2)=14: STAB(1,0)=100
2820 GRAPHICS 2+16
2830 FOR I1=1 TO 5
2840 SOUND 1,STAB(1,0),14,15
2850 FOR I2=1 TO 10:
     STAB(0,0)=STAB(0,0)-2:
     SOUND 0,STAB(0,0),14,STAB(0,2):
     NEXT I2
2860 STAB(0,0)=STAB(0,0)-20
2870 SOUND 1,0,0,0: SOUND 1,10,9,10
2880 SETCOLOR 4,RND(0)*14+1,10
2890 POSITION 3,4: PRINT #6;"SOUND-CHECK"
2900 NEXT I1
2910 SOUND 1,0,0,0
2920 SETCOLOR 0,1,9: SETCOLOR 1,9,12:
     SETCOLOR 2,9,12: SETCOLOR 4,4,1
2930 POSITION 13,5:PRINT #6,"produziert{10 SPACE}
     mit eurem atari{5 SPACE}den supersound"
2940 FOR I1=1 TO 14:STAB(0,2)=STAB(0,2)-1:
     SOUND 0,STAB(0,0),14,STAB(0,2):FOR I2=1 TO 3:
     NEXT I2:NEXT I1
2950 IF DISP>10 THEN DISP=0
2960 DISP=DISP+1
2970 POSITION 3,4: PRINT #6;"SOUND-CHECK"
2980 REM * LAUFEFFEKT
2990 POSITION 2+DISP,4:PRINT #6;"{INV(*)}"
3000 REM * VERLASSEN DES TITELBILDES
3010 IF STICK(0)<>15 THEN 3050
3020 IF STRIG(0)=0 THEN 3060
3030 FOR I1=15 TO 0 STEP -1:SOUND 0,10,10,I1:NEXT I1
3040 GOTO 2950
3050 IF STICK(0)<>15 THEN 3050
3060 IF STRIG(0)=0 THEN 3060
3070 REM INITIALISIEREN WERTETABELLE FUER GENERATOREN
3080 FOR I1=0 TO 3:FOR I2=0 TO 2:STAB(I1,I2)=0:
     NEXT I2:NEXT I1
3090 STAB(0,2)=8
3100 DIRE=0: IMPS=5
3110 RETURN
3200 REM * UNTERPROGRAMM ENDE
3210 GRAPHICS 2+16: SETCOLOR 4,4,1
3220 SOUND 1,0,0,0: SOUND 2,0,0,0: SOUND 3,0,0,0
3230 FOR I1=50 TO 0 STEP -1
3240 FOR I2=1 TO 2: NEXT I2
3250 SOUND 0,I1,14,8
3260 IF I1<12 THEN SETCOLOR 4,4,14-I1
3270 NEXT I1
3280 FOR I1=1 TO 30: NEXT I1: END
3290 REM ***  ENDE SOUNDCHECK *** ■
```

Tho.K.

FASTLOAD

Datenübertragung mit Schallgeschwindigkeit

Der große Nachteil bei der Speicherung von Daten auf Cassette oder Diskette in BASIC sind die langen Wartezeiten. Bei Daten, die mit GET oder PUT übertragen werden, kann man während des Schreib- oder Lesevorgangs Kaffee trinken gehen.

Ein Maschinenprogramm, das als ASCII-Zeichen in einer Stringvariablen abgelegt wird, kann Daten so schnell übertragen, wie ein BASIC-Programm geladen wird.

Der Datentransfer im Atari oder zwischen Atari und Peripheriegeräten wird von einer zentralen Routine im Betriebssystem gesteuert, der CENTRAL INPUT/OUTPUT Routine, kurz CIO. Diesem (Maschinen-) Unterprogramm wird mitgeteilt, welche Daten von wo nach wo geschickt werden sollen, den Rest erledigt der Computer selbständig. Die einzelnen Parameter werden in Speicherstellen bei $0340 (Dez. 832) bis $03BF (Dez. 959) abgelegt, den IOCBs (Input/Output Control Block). Es sind acht solcher IOCBs vorhanden, die vom BASIC bekannten Leitungen #0 bis #7. Der IOCB #0 wird immer durch den EDITOR (englisch, bedeutet Herausgeber) belegt. Hierüber werden Bildschirm-Aus- und Eingabe gesteuert. IOCB #6 wird für den Datenverkehr in den Graphistufen verwendet. Diese Datenkanäle dürfen nicht für andere Zwecke genutzt werden, da sonst eine Ausgabe von Daten an das TV-Gerät nicht mehr möglich ist. IOCB #7 wird bei Druckerfunktionen wie LPRINT oder LIST "P" belegt.

Jeder IOCB umfaßt 16 Bytes. Die Namen und Funktionen sind:

- **ICHID:** Zeigt an, welches Gerät den IOCB gerade benutzt. Der Wert 255 bedeutet „IOCB frei".
- **ICDNO:** Gerätenummer. Bei Disketten die Laufwerksnummer (DI-4).
- **ICCOM:** Kommando. Entspricht dem BASIC-Befehl 'XIO' (3 = OPEN, 12 = CLOSE usw.).
- **ICSTA:** Statusbyte. Enthält bei fehlerfreier Übertragung den Wert Eins, sonst den Fehlerkode.
- **ICBAL, ICBAH:** 16-Bit Startadresse des Datenpuffers oder bei OPEN-Funktion Zeiger auf den Filenamen (D:, C:).
- **ICPTL, ICPTH:** Zeiger auf die Routine, mit der einzelne Bytes übertragen werden. Wird vom Betriebsystem beim Öffnen eines Kanals gesetzt (Startadresse -1).
- **ICBLL, ICBLH:** Länge des Datenpuffers als 16-Bit Zahl. Bei Datenempfang wird hier die Anzahl der eingegangenen Bytes gespeichert.
- **ICAX1:** 1. Hilfsbyte. Gibt bei Eröffnung eines Datenkanals 2- die Richtung an (4:Eingabe, 8:Ausgabe).
- **ICAX2:** 2. Hilfsbyte. Für spezielle Funktionen, die von Device zu Device verschieden sind. Beim Cassettenrecorder bedeutet ein Wert über 127, daß der Motor zwischen zwei Datenblöcken nicht stoppt.
- **ICAX3 bis ICAX6:** Diese Bytes werden nur selten genutzt, z.B. bei NOTE oder POINT wird hier die Sektornummer angegeben.

Die Übertragung

Bevor ein Datensatz gelesen oder geschrieben werden kann, muß zum Gerät eine Leitung eröffnet werden. Unsere Routine könnten wir zwar hierfür nehmen, aber der BASIC OPEN-Befehl ist einfacher. Dazu eine Besonderheit: Um den Casettenrekorder im schnelleren Modus zu benutzen, wird (wie vorher schon erwähnt) die zweite Zahl im OPEN-Befehl mit 128 angegeben. Zum Laden eines, mit CSAVE abgelegten, Programms sieht das z.B. so aus: OPEN #1,4,128, "C". Bei einer Schreibfunktion mit Cassette sollte nach Öffnen der Leitung der Motor des Rekorders mit POKE 54018,60 abgeschaltet werden, da sonst beim Lesen Fehler auftreten könnten. Alle anderen Geräte werden normal eröffnet. Der Datensatz, der übertragen werden soll, wird am zweckmäßigsten in einem String untergebracht. Numerische Variablen könnt Ihr mit der STR$ oder CHR$ Funktion umsetzen, die Rückwandlung entsprechend mit VAL und ASC, wobei CHR$ nur Zahlen bis 255 verarbeitet. Verschiedene Zeichenspeicher werden in einem großen String zusammengefaßt. Eventuell sollte man Grenzzeichen einbauen, um die Teilstrings zu trennen. Wer schon mehr Erfahrung mit dem 'freien RAM' hat, kann seinen Datensatz auch hinter dem BASIC-Programm ablegen. Der 16-Bit Zeiger MEMTOP ($90,91/Dez.144,145) gibt das Ende des von BASIC belegtem RAMs an. Zu dieser Zahl sollte sicherheitshalber die Zahl 100 addiert werden, da bei einem Programmstopp (ob Fehler oder durch STOP) der nächste über die Tastatur eingegebene Befehl am Ende des Programms abgelegt wird. Dabei würden die Daten verlorengehen. Wenn der Datenpuffer soweit vorbereitet ist, wird die Maschinenroutine mit

X = USR (ADR(ML$), IOCB*16, ICCOM, ICBA, ICBL)

aufgerufen.

ADR (ML$) gibt die Startadresse der Subroutine an.

IOCB ist die Nummer der eröffneten Leitung.

ICCOM ist die Kommando-Kennzahl und identisch mit den Kennzahlen für den XIO Befehl.

ICBA ist die Startadresse des Datenpuffers. Bei Übertragung in oder aus einem String wird die ADR (...$) Funktion eingesetzt, bei freiem RAM die Adresse des ersten Datenbytes.

ICBL gibt die Länge des Datenblocks an. Theoretisch kann die Routine Datensatzlängen von 1 bis 65535 verarbeiten. Wird ein String übertragen, dann setzt hier LEN (...$) ein.

Verschiedene Datensätze können durch mehrfaches Aufrufen der Routine zu einem großen File zusammengefaßt werden, solange kein CLOSE Befehl gegeben wird. Sollen aber keine Daten mehr gespeichert werden, dann muß unbedingt ein CLOSE # Befehl folgen, damit der interne Datenpuffer geleert und übertragen wird!

Soll ein File geladen werden, so muß vorher der erforderliche Speicherplatz

reserviert werden. Wenn die Länge des Datensatzes bekannt ist, gibt es hierbei keine Probleme. Bei größeren Files mit unbekannter Länge könnt ihr folgendermaßen vorgehen: Einen String so groß wie möglich DIMensioniern (Bis zum Wert in FRE(0) ist das machbar!) — und dort die Daten einlesen. — Nach dem letzten Byte gibt das OS einen End Of File Fehler (EOF) aus. Moment mal, was ist 'OS'? Die Abkürzung für 'OPERATING SYSTEM', zu deutsch Betriebssystem (Ihr glaubt nicht, wie viele Leute dieses Kürzel schon gehört haben, aber seine Bedeutung nicht kennen!). Tritt nun beim Laden ein anderer Fehler auf, z.B. bei Cassettenbetrieb der berühmte ERROR 140 oder 143 AT LINE 0, so sind die Daten, die vorher gelesen wurden, im String abrufbar. Manches BASIC-File, das beim Laden schlicht streikte, konnte ich so rekonstruieren (Mehr über Atari-BASIC und Cassettenrecorder in einem der nächsten Hefte!). Die Anzahl der geladenen Bytes ist über die Speicherstellen ICBLL und ICBLH abfragbar. Nun muß dem BASIC noch gesagt werden, daß da irgend etwas in diesem String ist. Im Listing 1 geschieht dies in der Zeile 280. Hinter der letzten Stelle wird ein 'Null-String' gesetzt. Dadurch werden die Daten anerkannt und der Längen-Zeiger entsprechend gesetzt. Als Wert für ICBL muß beim Aufrufen der Routine die gleiche Zahl eingesetzt werden, auf die der String 'geDIMmt' wurde, dahinter kommt ein -1. Ein zu langes File würde sonst in den BASIC-Speicherbereich hinein abgelegt werden und das Programm zerstören. Reicht der Speicher nicht aus, so darf die INPUT-Leitung nicht geschlossen werden und das Subprogramm wird nach Bearbeitung des ersten Datensatzes erneut angesprungen. Die Status- (Fehler) Meldung 1 besagt, daß die Routine fehlerfrei gearbeitet hat. Weitere Daten sind abrufbar.

Laden eines Files

Das Listing

In Zeile 150 wird die Stringvariable DAT$ dimensioniert. Dabei muß beachtet werden, daß in der Subroutine bei 500 noch ein String (ML$) hinzukommt. Bei Recorder-Betrieb muß in Zeile 190 bzw. 220 die Betriebsart angegeben werden, um die Datenleitungen entsprechend zu eröffnen. Die Eingaben werden anschließend zur Kontrolle nochmals angezeigt und müssen mit 'Y' (für Yes) bestätigt werden. Nach Öffnen der Leitung, Übergabe der Parameter und Aufrufen der Subroutine wird in Zeile 290 die Status-Variable STAT ausgewertet. Wurde als Device der Recorder in schneller Betriebsart angesprochen und ist das Datenfile länger als der Datenpuffer, dann erscheint auf dem Bildschirm die Meldung 'File zu lang...' und das Programm endet. Durch das Stoppen des Bandes ist eine fehlerfreie Übertragung nicht mehr möglich.

Bevor der Datenpuffer wieder ausgegeben wird, wartet das Programm in Zeile 330 auf Tastendruck. Jetzt kann, wenn auf das gleiche Device gesendet werden soll, die Cassette oder Diskette gewechselt werden. Die Initialisierung der Subroutine und deren Aufruf erfolgen in den Zeilen 340 bis 360. Ein erneutes Öffnen der Ausgabeleitung wird durch die Zeiger-Variable O2 verhindert; ist ihr Wert ungleich Null, dann erfolgt ein Sprung in Zeile 350. Eine Eins in der Variablen C besagt, daß noch mehr Daten zu übertragen sind. Dann springt das Programm von 390 auf 270.

In den Zeilen 500 bis 560 wird das Maschinenprogramm in ML$ abgelegt. Die nötigen Parameter wurden im vorangegangenen Text bereits erklärt. In Zeile 590 und 620 werden die Rückmeldungen in STAT für STATUS und L wie Länge abgelegt. Ihr könnt das Subprogramm von Zeile 500 bis 620 ohne Änderung in euer Programm einbauen. Das TRAP-Statement in Zeile 510 bewirkt, daß ML$ nur beim ersten Aufruf initialisiert wird. Die in den DATA-Zeilen 530 bis 560 liegenden ML-Codes für das Maschinenprogramm werden in den Zeilen 510 und 520 in den String ML$ geladen. Der Sprung in die ML-Routine wird in Zeile 580 ausgeführt, die Rückmeldungen in den Zeilen 590 bis 610 übernommen und auf Fehlercode geprüft. Bei einem Fehler wird die entsprechende Nummer angezeigt und das Programm abgebrochen.

Das Listing des Maschinenprogramms soll hier nicht weiter erklärt werden, da dieser Artikel dann das halbe Heft füllen würde. Die Ansteuerung der CIO in Maschinensprache wird ein in sich geschlossener Beitrag. Mit dieser BASIC-Routine können bei 48 Kbyte und geladenem DOS ca. 27000 Bytes in einem Durchgang geladen werden. Das entspricht 216 Disketten-Sektoren.

```
100 REM FASTLOAD Vers. 1.2
110 REM Ein universelles
120 REM Kopierprogramm
130 REM (C) 1984 by Gunnar Binder
140 DIM FILE1$(20),FILE2$(20):POKE 82,0
150 RAM=FRE(0)-300:DIM DAT$(RAM+1)
160 ? "{CLEAR}{2 TAB}FASTLOAD":
    ? "{3 SPACE}INPUT-FILE ";
170 INPUT FILE1$:OPEN #5,4,0,"K"
180 IF FILE1$(1,1)<>"C" THEN 200
190 ? "{INV(N)}ormaler oder {INV(S)}chneller
    Modus?";:GET #5,X:? CHR$(X):AUX1=(X=83)*128
200 ? :? "{3 SPACE}OUTPUT-FILE ";:INPUT FILE2$
210 IF FILE2$(1,1)<>"C" THEN 230
220 ? "{INV(N)}ormaler oder {INV(S)}chneller
    Modus?";:GET #5,X:? CHR$(X):AUX2=(X=83)*128
230 POSITION 2,1:
    ? "{7 DEL.LINE}{DOWN}{3 SPACE}Zur Kontrolle:"
240 ? "Input{2 SPACE}:";FILE1$:? "Output :";FILE2$:
    ? :? "Ist das ok? (Y/N).{LEFT}";
250 GET #5,X:ON (X<>89 AND X<>78) GOTO 250:
    POKE 752,1:? CHR$(X):IF X=78 THEN RUN
260 CLOSE #1:OPEN #1,4,AUX1,FILE1$
270 IOCB=1:ICCOM=7:ICBA=ADR(DAT$)
```

```
280 ICBL=RAM:GOSUB 500:C=0:DAT$(L+1)=""
290 ON STAT=136 GOTO 310:IF AUX1 OR AUX2 THEN ? "File
    zu lang, kann nicht kopiert werden.":END
300 ? "Datenblock geladen.":C=1:GOTO 320
310 ? "File komplett geladen"
320 ? "Datenblock ist ";L;" Bytes lang":
    ? "Ausgabe mit {INV( RETURN )} starten ";
330 GET #5,X:POKE 752,1:? :IF 02 THEN 350
340 CLOSE #2:OPEN #2,8,AUX2,FILE2$:02=1
345 IF FILE2$(1,1)="C" THEN POKE 54018,60
350 IOCB=2:ICCOM=11:ICBA=ADR(DAT$)
360 ICBL=L:GOSUB 500:POKE 752,0
370 IF  NOT C THEN CLOSE #2:
    ? "Kopiervorgang beendet":END
380 ? "Datenblock geschrieben.":
    ? "Weiter mit {INV( RETURN )}.";
390 GET #5,X:POKE 752,1:? :GOTO 270
500 REM < Maschinenroutine in ML$ >
510 TRAP 570:DIM ML$(28):TRAP 40000:FOR I=1 TO 28
520 READ X:ML$(I,I)=CHR$(X):NEXT I
530 DATA 104,104,104,170,104,104
540 DATA 157,66,3,104,157,69,3
550 DATA 104,157,68,3,104,157,73
560 DATA 3,104,157,72,3,76,86,228
570 REM < CIO aufrufen >
580 X=USR(ADR(ML$),IOCB*16,ICCOM,ICBA,ICBL)
590 L=PEEK(840+IOCB*16)+256*PEEK(841+IOCB*16)
```

```
600 STAT=PEEK(835+IOCB*16)
610 IF STAT=1 OR STAT=136 THEN RETURN
620 ? "{INV( Fehler )} ";STAT:END
```

```
        0100 IOCB    =   $340   Maschinensprache
        0110 ICCOM   =   IOCB+2
        0120 ICBAL   =   IOCB+4
        0130 ICBAH   =   IOCB+5
        0140 ICBLL   =   IOCB+8
        0150 ICBLH   =   IOCB+9
        0160 CIOV    =   $E456
68      0170         PLA
68      0180         PLA
68      0190         PLA
AA      0200         TAX
68      0210         PLA
68      0220         PLA
9D4203  0230         STA  ICCOM,X
68      0240         PLA
9D4503  0250         STA  ICBAH,X
68      0260         PLA
9D4403  0270         STA  ICBAL,X
68      0280         PLA
9D4903  0290         STA  ICBLH,X
68      0300         PLA
9D4803  0310         STA  ICBLL,X
4C56E4  0320         JMP  CIO
        0330         .END
```

ACHTUNG: Ab 31. 03. 84 neue Adresse!

Preissenkung der Software zur Neueröffnung !!!

Umfangreiche Software für ATARI u. Commodore vorhanden! Fordern Sie unsere Info- und Sonderangebotsliste an. Hier ein Auszug aus unserer Software-Preisliste:

SOFTWARE-PAKETPREIS: 3 Programme nach Ihrer Wahl (je 1 Programm der Softwaregruppen A, B und C aus unserer umfangreichen Softwareliste) zum Preis von: **175,–**

UNSERE TOP-PROGRAMME

S. A. M. läßt Ihren ATARI sprechen! 119,–
ARCADE MACHINE erzeugt Spielprogr. in Masch.-Sprache 155,–
PINBALL CONSTR. SET 115,–
DARK CRYSTAL 109,–
CASTLE WOLFENSTEIN 89,–
ZAXXON (3-D GRAPH.) 98,–
PINBALL NIGHTM. 89,–
THE PHARAOH'S CURSE 89,–
FORT APOCALYPSE 89,–
BANDITS 89,–
AXIAS ASSISSIN 115,–
CHESS 7.0 165,–
WAY OUT 89,–
CRISIS MOUNTAIN 89,–
CYTRON MASTER 89,–
M.U.L.E. 110,–
ARCHON 110,–
HARD HAT MACK 110,–
TALE OF BETA LYRAE 98,–
SOLO FLIGHT 98,–
ENCOUNTER (BAT. ZONE) 98,–
DIMENSION X 98,–
NATO COMMANDER 98,–
MIG ALLEY ACE 98,–
A. E. (3-D GRAPHIC) 98,–
FLIGHT SIMULAT. II 115,–
BLUE MAX (3-D GR.) 98,–
JUMPMAN 98,–
ZEPPELIN 98,–
LODE RUNNER 98,–
SHADOW WORLD 89,–
REPTON 89,–
OPERATION WHIRLWIND 98,–
POOYAN 89,–

ATARI SOFTWARE A

SKY BLAZER 85,–
THRESHOLD 85,–
NECROMANCER 85,–
JEEPERS CREEPERS 85,–
NIGHT RAIDERS 85,–
TIME RUNNER 85,–
PREPPIE I 85,–
CHOPLIFTER 85,–
DAVID'S MID. MAGIC 85,–
WAVY NAVY 85,–
DRELBS 85,–
WIZARD OF WOR 85,–
CHOPPER RESCUE 85,–
THE ELEMINATOR 85,–
COHEN'S TOWER 85,–
FLIP FLOP 85,–
MR. ROBOT a. h. FACT. 85,–
SHAMUS CASE II 85,–
BRISTLES 85,–
CAPTURE THE FLAG 85,–
COSMIC TUNNEL 85,–
QUASIMODO 85,–
SLIME 85,–
SEA FOX 85,–
ZOMBIES 85,–
SURVIVOR 85,–
SNOKIE 85,–

ATARI SOFTWARE B

MOUSKATTACK 75,–
SHOOTOUT 75,–
PROTECTOR II 75,–
RASTER BLASTER 75,–
MARAUDER 75,–
THE ELEMINATOR 75,–
CROSSFIRE 75,–
FINAL FLIGHT 75,–
TWERPS 75,–
O'RILEYS MINE 75,–
FROGGER 75,–
TUMBLE BUGS 75,–
ROBOT BATTLE 75,–
B.C.'S QUEST TIRES 75,–
ROUNDABOUT 75,–
JUICE 75,–
MONSTER SMASH 75,–
CLAIM JUMPER 75,–
AIRRAID 75,–
TRION 75,–
SARGON II (SCHACH) 75,–
PREPPIE II 75,–
WALL WAR 75,–
NAUTILUS 75,–
LUNAR LEEPER 75,–
ARENA 3000 75,–
STRIP-POKER ab 18 J. 75,–

ATARI SOFTWARE C

ANT-EATER 65,–
COLLISION COURSE 65,–
CANYON CLIMBER 65,–
SENTINEL 65,–
MOON PATROL 65,–
SPY'S DEMISE 65,–
KID GRID 65,–
ROSENS BRIGADE 65,–
MAD NETTER 65,–
JAWBREAKER (PACMAN) 65,–
SEA DRAGON 65,–
GRIDRUNNER 65,–
SNAKE BYTE 65,–
CYCLOD 65,–
EIS HOCKEY 65,–
GHOST HUNTER 65,–
SNAPPER 65,–
STELLAR SHUTTLE 65,–
MEGALEGS 65,–
PICNIC PARANOIA 65,–
PROTECTOR I 65,–
GAL. CHASE (GALAXIAN) 65,–
APPLE PANIC 65,–
BUG ATTACK 65,–
SNEAKERS 65,–
GOLF CHALLENGE 65,–

C-64 SOFTWARE

S. A. M. läßt Ihren C-64 sprechen! 119,–
ZAXXON 98,–
FORT APOCALYPSE 89,–
SURVIVOR 79,–
JUMPMAN 89,–
REPTON 89,–
PHARAOS CURSE 89,–
SQUISH'EM 69,–
D. MIDNIGHT MAGIC 79,–
POOYAN 89,–
BANDITS 89,–
DRELBS 89,–
FROGGER 89,–
WAY OUT 89,–
O'RILEY'S MINE 89,–
SWASHBUCKLER 89,–
SEA DRAGON 79,–
ROBOT BATTLE 79,–
SHAMUS 79,–
BLADE OF BLACKPOOLE 89,–
SEAFOX 79,–
JAWBREAKER 89,–
BENJI SPACE RESCUE 96,–
TEMPLE OF APSHAI 89,–
ZORK I, II, III je 98,–
BLUE MAX (3-D) 98,–
CASTLE WOLFENSTEIN 89,–
BEACH HEAD 89,–
SOLO FLIGHT 98,–
PINBALL NIGHTM. 89,–
FLIGHTSIMULATOR II 119,–
FAST EDDIE 79,–
MIG ALLEY ACE 98,–
PINBALL CONSTR. SET 119,–
HARD HAT MACK 110,–
ARCHON 115,–
M.U.L.E. 115,–
DIMENSION X 98,–
MOON SHUTTLE 69,–
PROTECTOR 79,–
SNOKIE 79,–
LODE RUNNER 89,–
CHESS 7.0 165,–
CRITICAL MASS 79,–
B.C.'S QUEST TIRES 79,–
WIZARD OF WOR 89,–
CROSSFIRE 79,–

weitere Programme siehe Katalog!

ZENITH-Monitor, blendfr.	**298,–**
COMMODORE 64, 64 KB-RAM	**745,–**
COMMODORE 64 + DISK-DRIVE	**1498,–**
VC 1526, MATRIX-DRUCKER	**848,–**
VC 1541 DISK-DRIVE	**795,–**
VC 1530, DATASETTE	**135,–**
ATARI 600 XL, 16 KB, Basic	**545,–**
ATARI 800 XL, 64 KB, Basic	**895,–**
ATARI 1010 Recorder	**245,–**
ATARI 1050 DISK-DRIVE	**995,–**
16 KRAM für ATARI 800	**129,–**
48 KRAM für ATARI 400	**298,–**
ATARI-CENTRONICS-Interface, wird über den seriellen Ausgang betrieben!!!	**255,–**
ATARI-Kass.-Rec.-Interface	**119,–**
ATARI 400, 16 KB-RAM, BASIC	**398,–**
Joystick, einfach	**35,–**
Joystick 'COMPETITION PRO'	**69,–**
WICO-Joystick	**ab 78,–**
Light-Pen f. C-64 + ATARI	**129,–**
KOALA-PAD f. C-64 + ATARI	**335,–**
STAUBSCHUTZHÜLLEN für: COMMODORE + ATARI	**ab 26,–**

Alle Preise verstehen sich in DM, inkl. MwSt.! Versand erfolgt gegen Vorkasse oder Nachnahme nach unseren allgemeinen Lieferbedingungen. Fordern Sie spez. Info gegen 2,– DM in Briefmarken an. Die Angebote gelten solange der Vorrat reicht.

Programmier-Sprachen (MICROSOFT-BASIC, BASIC-A +, SIMON'S BASIC, ASSEMBLER, FORTH, PASCAL, PILOT, BASIC-COMPILER, usw.), DOS, Utilities (Anwender-Programme), Graphic-Hilfsprogramme usw. für ATARI u. COMMODORE lieferbar!!!)

VECOS-WARNECKE
Hildesheimer Str. 52 · 3000 Hannover 1

Öffnungszeit von 9.00 bis 18.30 Uhr, von 10.00 bis 20.00 Uhr telefonisch erreichbar (05 11) 88 27 99

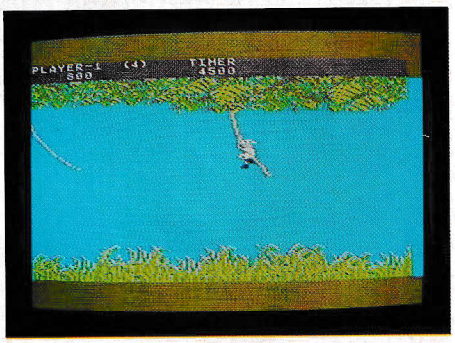
SOFTWARE IM TEST

Jungle hunt

Es ist unschwer zu erkennen, daß Jungle Hunt die Heimcomputer-Version des Arkadespiels Jungle King ist. Für alle, die bisher noch nicht das Vergnügen hatten, mit dem Helden dieses Spiels Bekanntschaft zu schließen, sei der Spielverlauf kurz erklärt.

Man beginnt dieses Spiel auf einem Ast, der als Sprungbrett zu Ruhm, Ehre und vielen Punkten dienen soll. Vor einem schwingen Lianen, an denen man sich bis zu einem Fluß hangeln muß. Darin warten allerdings schon gierige Krokodile mit leerem Magen. Hat man dieses Gefahrenquelle durchschwommen, kommt man an einen Berg, der erstürmt werden will. Keine große Schwierigkeit, wären da nicht Felsbrocken, die einem entgegenpoltern und das Erklimmen des Hanges erschweren.

Ist auch diese Aufgabe bewältigt, erkennt man endlich, warum man alle diese Strapazen auf sich genommen hat: Mit Entsetzen erblickt man ein weibliches Wesen, das über einem gut geheizten Kochtopf hängt. Der Apell an die Ritterlichkeit des Spielers ist unmißverständlich: nichts wie hin und sie retten! Doch so einfach ist das nicht, denn vor einem tanzen speerschwingende Menschenfresser, die offensichtlich nicht geneigt sind, das Mädchen ohne weiteres rauszurücken. Diese Eingeborenen sollen nun aber nicht brutal niedergemetzelt werden, nein, man muß mit einem gewaltigen Sprung über sie wegsetzen.



Ist auch diese Hürde genommen, befindet man sich — zumindest für kurze Zeit — am Ende seiner Mission. Aber gleich darauf beginnt, für besonders mutige und ritterliche Tarzäne, eine zweite, erschwerte Errettung.

Junge Hunt ist, vor allem wegen seiner sehr guten Graphik, eine gelungene Kopie vom Spielhallen-Original Jungle King. Der Spielwitz ist bei dieser Kopie keineswegs auf der Strecke geblieben, sondern wurde, anders als bei vielen anderen Beispielen, voll auf den heimischen Bildschirm übertragen. Wer ein Freund von gut gemachten Fun-Games ist und sich nicht an der Tatsache stört, daß sich Tarzan in dieser Version in ein niedliches Kerlchen mit Tropenhelm verwandelt hat, dem können wir dieses Spiel wärmstens empfehlen.

Graphik:

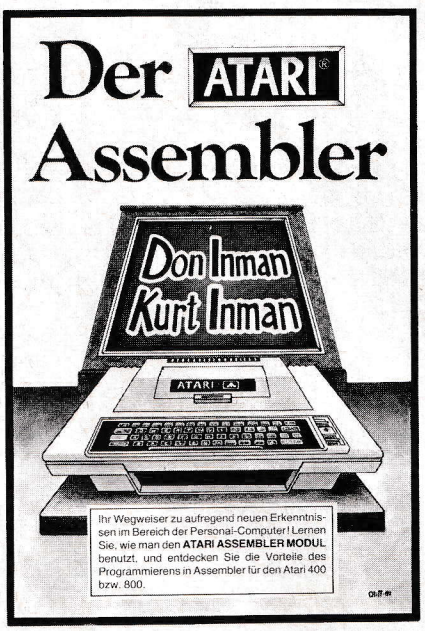
Sound:

Spielidee:

Testergebnis:

Lohnend

Dieses Buch vermittelt alles Wissen, das nötig ist, um mit dem Atari-Assembler zu programmieren. Vorausgesetzt wird der Atari-Assembler-Modul, aber weder in Assembler noch in Maschinensprache muß man Vorkenntnisse haben. Das BASIC aber sollte gut sitzen.



Der ATARI-Assembler
Inman/Inman
IDEA Verlag GmbH
35,— DM
268 Seiten
ISBN 3-88793-025-8

Die Autoren führen den Leser über BASIC an die Maschinenprogrammierung heran und zeigen dann, wie der Assembler diese unterstützt. Es gelang, wie im Vorwort versprochen, das Buch leicht lesbar und verständlich zu machen. Durch Grafiken und klares Schriftbild, Tabellen und Register wird das Lernen wesentlich erleichtert.

Fazit: Dieses ist **das** perfekte Buch zum Atari, man fragt sich, warum es nicht von Atari als zusätzliches Handbuch verkauft wird. Maschinensprache-Einsteigern ist die Investition unbedingt zu empfehlen, die Anschaffung lohnt aber sicher auch für den in Maschinensprache und Assembler Erfahrenen.

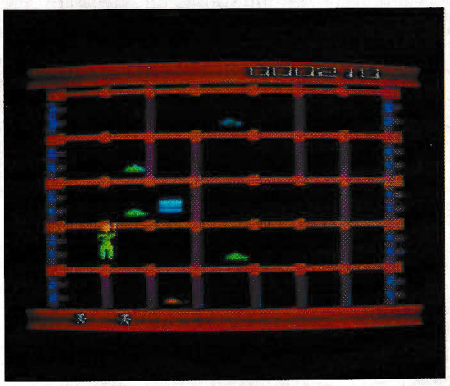
Ebenfalls erfreulich ist der im Verhältnis zur Qualität doch recht bescheidene Preis.

Thok

Squish'em

Squish'em ist ein Wort, das wohl aus der englischsprachigen Comicliteratur kommt und soviel bedeutet, wie 'Zermansche sie'. In diesem Spiel ist eine Trägerkonstruktion zu erklimmen, auf deren Spitze Reichtum den kühnen Erstürmer erwartet. Hinterlistige Monster versuchen jedoch, den Spieler am Erfolg zu hindern, indem sie ihn mit Gegenständen bewerfen und seine Schuhsohlen anknabbern. Ihnen gilt es zu entkommen oder einen gezielten Tritt auf's Haupt zu verpassen.

Schaltet man seinen Computer ein, so kommt das Gerät, ohne ein Anfangsbild zu übertragen, direkt zur Sache. Die Graphik, die stattdessen auf dem Bildschirm sofort erscheint, ist sehr phantasiereich gestaltet. Und gerade, wenn der Spieler zutritt, hat er die Lacher auf seiner Seite. Die Farbe des Bildschirmes ändert sich mit jeder Stufe, die man er-



klimmt. Auch die Monster werden zunehmend arglistiger und nehmen zuweilen bizarre Formen an.

Im Gegensatz zu vielen anderen Spielen steigert sich die Lust an Squish'em mit seiner Dauer.

Squish'em ist kein typisches 08/15 Spiel, sondern vielmehr ein lustig gemachtes Kletterspiel für all diejenigen, die Spielwitz über den action-Gehalt stellen. Den Fernsehton sollte man bei diesem Spiel allerdings abstellen und stattdessen lieber eine Platte auflegen. Der Sound, der zu Anfang noch lustig wirkt, wird nach spätestens 20—30 Minuten zur Qual.

Graphik:

Sound:

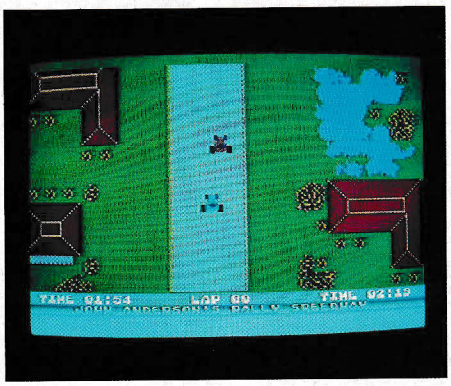
Spielidee:

Testergebnis:

Rallyeracer

Rallyeracer ist, wie der Name vermuten läßt, die Simulation eines Rallyerennens. Die Variationsmöglichkeiten bei diesem Spiel sind einfach zu beschreiben, denn sofort nachdem man dieses Spiel gestartet hat, erscheint auf dem Bildschirm eine umfangreiche Wahltafel auf der man Variablen einstellen kann, wie Straßenzustand (trocken, naß, vereist), Höchstgeschwindigkeit (40—100 km/h) und Anzahl der Spieler (1 oder 2). Hat man seine Wahl getroffen, erscheint auf dem Bildschirm in Draufsicht ein Teil einer Rennstrecke und das eigene und das gegnerische Auto, sofern man zu zweit spielt. Haben nun beide Spieler einmal den Joystick bewegt, geht die gelbe Post auch schon ab. Die Rennstrecke scrollt von nun an unter den rivalisierenden Rallyewagen hinweg. Ziel ist es, den gegnerischen Wagen um einen Bildschirmausschnitt hinter sich zu lassen (5 Strafsekunden für den Langsameren), oder den Gegner an eine der zahlreichen Häuserwände oder ein Waldstückchen zu drücken (10 Strafsekunden). Aber auch ohne einen gehässigen Gegner, d.h. im Alleinbetrieb, fällt es bei entsprechender Schwierigkeitsstufe schon enorm schwer, nicht an irgendwelche der vielen Hindernisse zu stoßen. Nach jeweils drei Runden ist ein Rennen beendet und man kann sofort ein neues Spiel wählen.

Beeindruckt hat uns an diesem Spiel vor allem die hervorragende Darstellung von Häusern, Swimmingpools und Garageneinfahrten, an denen der Spieler im



Laufe der mörderischen Rallyefahrt vorbeirast. Allgemein kann man diese Graphik, nicht zuletzt wegen der guten Farbwahl und der Originaltreue, guten Gewissens als sehr gut bezeichnen. Der Sound mag da in nichts nachstehen, denn auch er ist vom Quietschen der Reifen bis hin zum Aufprallgeräusch (bei unvorsichtigem Fahren) in allen Belangen realistisch und verleitet, was nur bei wenigen Spielen möglich ist, das Spiel mit der Spacetaste zu unterbrechen, einfach um den Fernsehton lauter zu stellen. Er erscheint bei diesem Spiel beinahe selbstverständlich, daß man nicht nur diesen einen im Programm abgespeicherten Rennstreckenverlauf spielen, sondern sich auch selbst Rennstrecken ausdenken, erstellen und abspeichern kann.

Alles in allem ist Rallyeracer ein Spiel, das man als einzelner Spieler haben sollte, und als einer von zwei etwa gleichwertigen Spielern unbedingt haben muß, denn selbst nach längerem Spiel, gerade zu zweit, erscheint es und durch die vielen Variationsmöglichkeiten beinahe unmöglich, daß Langeweile aufkommt.

Graphik:

Sound:

Spielidee:

Testergebnis:

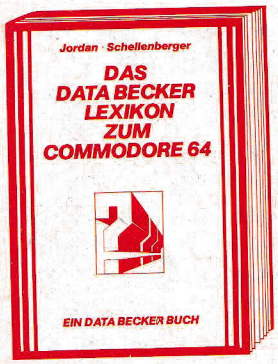
BUCHHITS VON

Das PASCAL-TRAININGSBUCH bietet nicht nur eine leicht verständliche Einführung in die Sprache PASCAL, sondern auch in die Technik des strukturierten Programmierens. Dabei wird der Befehlssatz des UCSD-PASCAL ebenso wie der des PASCAL 64-Compilers, der von DATA BECKER vertrieben wird, erläutert. Der schrittweise Aufbau des Buches, vom Einfachen zum Schwierigen, trägt zum guten Verständnis des PASCAL-Konzeptes bei; Übungen helfen dabei den Lernerfolg zu sichern. Das PASCAL-TRAININGSBUCH sollte nicht nur der C-64 Besitzer haben, sondern jeder, der endlich PASCAL beherrschen will.
PASCAL-TRAININGSBUCH, 1984,
ca. 250 Seiten, DM 39,—.

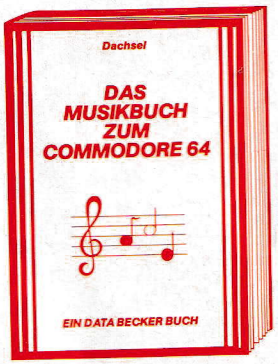
MULTIPLAN gehört zu den erfolgreichsten Kalkulationsprogrammen überhaupt und ist seit kurzem auch für den C-64 verfügbar. Das neue Trainingsbuch bietet eine Einführung in die Grundbegriffe der Tabellenkalkulation und erleichtert es dem MULTIPLAN-Einsteiger wesentlich, den umfangreichen Befehlssatz auch kommerziell zu nutzen. Dabei ermöglichen Übungen am Ende jedes Kapitels eine Vertiefung des erworbenen Wissens. Durch die vielen praktischen Anwendungsbeispiele bietet das MULTIPLAN-TRAININGSBUCH auch dem Fortgeschrittenen eine echte Hilfe.
MULTIPLAN-TRAININGSBUCH, 1984,
ca. 250 Seiten, DM 49,—.

Achtung Hobbyelektroniker: Diese Buch enthält nicht nur alles über Interfaces und Ausbaumöglichkeiten des COMMODORE 64, sondern auch über seine vielfältigen Einsatzmöglichkeiten von der Lichtorgel über Motorsteuerung, Spannungs- und Temperaturmessung bis zur programmierbaren Stromversorgung, und wie man diese verwirklicht. Zehn komplette Schaltungen zum Selberbauen, vom Eprommer über Eprom-Karte, Logic Analyzer, Freuqenzzähler, Hardware-Tracer, Pulsmeßgerät, Klatschschalter und Digital-Voltmeter bis zur preiswerten Spracheingabe-Sprachausgabe. Jeweils komplett mit Schaltplan, Layout und Softwarelisting.
Ca. 220 Seiten, DM 49,–, ab April 84.

SIMON's BASIC ist ein Hit – wenn man es richtig nutzen kann. Deshalb gibt es jetzt zu dieser vielseitigen Befehlserweiterung unser umfangreiches Trainingsbuch, das Ihnen detailliert den Umgang mit den über 100 Befehlen des SIMON's BASIC erklärt. Ausführliche Darstellung aller Befehle (auch der, die nicht im Handbuch stehen!) Natürlich auch mit allen Macken und Hinweisen, wie man diese umgeht. Dazu zahlreiche Beispielprogramme und interessante Programmiertricks. Nach jedem Kapitel Testaufgaben zum optimalen Selbststudium. Dieses Buch sollte jeder SIMON's BASIC Anwender unbedingt haben! Das TRAININGSBUCH ZUM SIMON's BASIC, 1984,
ca. 300 Seiten, DM 49,–.

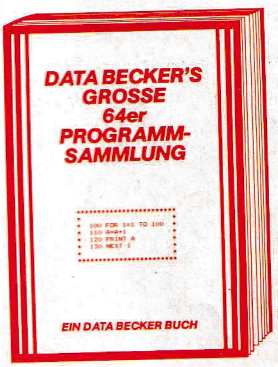


So etwas haben Sie gesucht:
Umfassendes Nachschlagewerk zum COMMODORE 64 und seiner Programmierung. Allgemeines Computerlexikon mit Fachwissen von A–Z und Fachwörterbuch mit Übersetzungen wichtiger englischer Fachbegriffe – das DATA BECKER LEXIKON ZUM COMMODORE 64 stellt praktisch drei Bücher in einem dar. Es enthält eine unglaubliche Vielfalt an Informationen und dient so zugleich als kompetentes Nachschlagewerk und als unentbehrliches Arbeitsmittel. Viele Abbildungen und Beispiele ergänzen den Text. Ein Muß für jeden COMMODORE 64 Anwender.
Ca. 350 Seiten, DM 49,–.



Der COMMODORE 64 ist ein Musikgenie, und mit diesem Buch lernen Sie alles über seine musikalischen Fähigkeiten. Der Inhalt reicht von einer Einführung in die Computermusik über die Erklärung der Hardware-Grundlagen und die Programmierung in BASIC bis hin zur fortgeschrittenen Musikprogrammierung. Zahlreiche Beispielprogramme und leicht verständliche Darstellung. Geschrieben vom Autor der bekannten Musikprogramme SYNTHIMAT und SYNTHESOUND.
Erschließen Sie sich die Welt des Sounds und der Computermusik mit dem MUSIKBUCH ZUM COMMODORE 64.
Ca. 200 Seiten, DM 39,–.

Das sollte Ihr erstes Buch zum COMMODORE 64 sein: 64 FÜR EINSTEIGER ist eine sehr leicht verständliche Einführung in Handhabung, Einsatz, Ausbaumöglichkeiten und Programmierung des COMMODORE 64, die keinerlei Vorkenntnisse voraussetzt. Schritt für Schritt führt das Buch Sie in die Programmiersprache BASIC ein, wobei Sie nach und nach eine komplette Adressenverwaltung erstellen, die Sie anschließend nutzen können. Zahlreiche Abbildungen und viele Anregungen zum sinnvollen Einsatz des COMMODORE 64.
Das Buch ist sowohl als Einführung als auch als Orientierung vor dem 64er Kauf gut geeignet. Ca. 200 Seiten, DM 29,–.



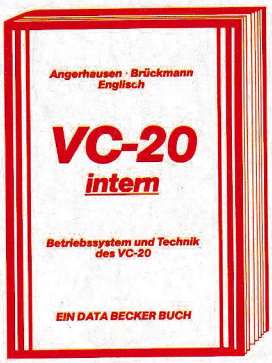
Diese neue, umfangreiche Programmsammlung hat es in sich. Über 50 Spitzenprogramme für den COMMODORE 64 aus den unterschiedlichsten Bereichen, vom Superspiel („Senso", „Pengo") über Graphik- und Soundprogramme (zum Beispiel „Fourier 64" oder „Orgel") sowie Utilities („Sort") bis hin zu Anwendungsprogrammen wie „Videothek" oder „Finanzbuchhaltung". Der Hit sind zu jedem Programm aktuelle Programmiertips und Tricks der einzelnen Autoren zum Selbermachen. Also – nicht nur abtippen, sondern auch dabei lernen und wichtige Anregungen für die eigene Programmierung sammeln.
Ca. 250 Seiten, DM 49,–.

Darauf haben Sie bestimmt gewartet

Die neue DATA WELT ist eine Computerzeitschrift speziell für COMMODORE-Anwender. Brandaktuell (detaillierte Informationen über die neuen COMMODORE Computer 264 und 364) und randvoll mit Berichten, Trends und interessanten Programmiertips. 80 Seiten stark im Magazinformat. Gleichzeitig als Nachfolger des VC-Infos umfassende Übersicht über aktuelle Produkte, Bücher und Programme rund um COMMODORE 64 und VC-20. Die Frühjahrsausgabe der neuen DATA WELT erhalten Sie ab Anfang März überall dort, wo es DATA BECKER BÜCHER und -PROGRAMME gibt. Am besten gleich holen oder direkt bei DATA BECKER gegen DM 4,– in Briefmarken anfordern.

DATA BECKER

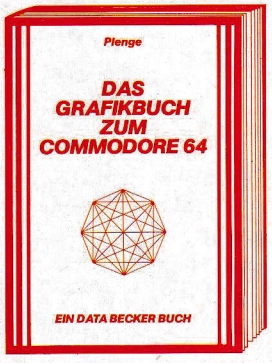
Die überarbeitete und erweiterte 2. Auflage von 64 TIPS & TRICKS enthält eine umfangreiche Sammlung von POKE's und anderen nützlichen Routinen, Multitasking mit dem C-64, hochauflösende Graphik und Farbe für Fortgeschrittene, mehr über CP/M auf dem C-64, mehr über Anschluß- und Erweiterungsmöglichkeiten durch USER PORT und EXPANSION PORT, sowie zahlreiche ausführlich dokumentierte Programme von der SORT-Routine über zahlreiche BASIC-Erweiterungen bis hin zur 3D-Graphik (alle Maschinenprogramme jetzt mit BASIC-Ladeprogramm!). 64 TIPS UND TRICKS ist eine echte Fundgrube für jeden COMMODORE 64 Anwender.
64 TIPS & TRICKS, 2. Auflage 1983, ca. 290 Seiten, DM 49,–.



Die überarbeitete und erweiterte 2. Auflage von VC-20 INTERN beschäftigt sich detailliert mit Technik und Betriebssystem des VC-20 und enthält ein ausführlich dokumentiertes ROM-Listing, die Belegung der ZEROPAGE und anderer wichtiger Bereiche, übersichtliche Zusammenfassungen der Routinen des BASIC-Interpreters und des VC-20 Betriebssystems, eine Einführung in die Programmierung in Maschinensprache, eine detaillierte Beschreibung der Technik des VC-20 und als Clou drei Original COMMODORE Schaltpläne zum Ausklappen! Damit ist VC-20 INTERN für jeden interessant, der sich näher mit Technik und Maschinenprogrammierung des VC-20 auseinandersetzen möchte. VC-20 INTERN, 2. Auflage 1983, ca. 230 Seiten, DM 49,–.

Jetzt in überarbeiteter und erweiterter 3. Auflage: 64 INTERN erklärt detailliert Architektur und technische Möglichkeiten des C-64, zerlegt mit einem ausführlich dokumentierten ROM-Listing Betriebssystem und BASIC-Interpreter, bringt mehr über Funktion und Programmierung des neuen Synthesizer Sound Chip und der hochauflösenden Graphik, zeigt die Unterschiede zwischen VC-20, C-64 und CBM 8000 und gibt Hinweise zur Umsetzung von Programmen. Zahlreiche lauffertige Beispielprogramme, Schaltbilder und als Clou: zwei ausführlich dokumentierte Original COMMODORE Schaltpläne zum Ausklappen. Dieses Buch sollte jeder 64-Anwender und Interessent haben. 64 INTERN, 3. Auflage 1983, ca. 320 Seiten, DM 69,–.

neu **Die Diskette zum Buch mit allen wichtigen Programmen DM 39,-**



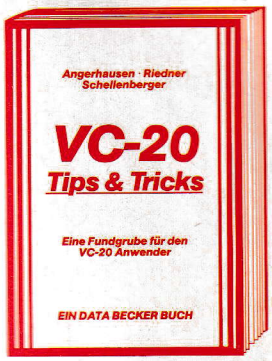
Graphik ist eine der Hauptstärken des COMMODORE 64. Mit diesem neuen Buch lernen Sie, wie Sie die graphischen Fähigkeiten programmtechnisch optimal nutzen. Der Inhalt reicht von den Grundlagen der Graphikprogrammierung über das Erzeugen einfacher Figuren, die Arbeit mit Sprites, Zeichensatzprogrammierung, Hardcopy und IRQ-Handhabung bis hin zur Funktionendarstellung, Laufschrift, Statistik, 3-D, CAD, den Geheimnissen der Actionsspiele und Lightpenanwendungen. Zahlreiche Beispielprogramme ergänzen dieses Buch, das die faszinierende Computertechnik jedermann zugänglich macht.
Ca. 250 Seiten, DM 39,–.

neu **Die Diskette zum Buch mit allen wichtigen Programmen DM 39,-**

Wer besser und leichter in BASIC programmieren möchte, der braucht dieses neue Buch. 64 FÜR PROFIS zeigt, wie man erfolgreich Anwendungsprobleme in BASIC löst und verrät Erfolgsgeheimnisse der Programmierprofis. Vom Programmentwurf über Menüsteuerung, Maskenaufbau, Parameterisierung, Datenzugriff und Druckausgabe bis hin zur Dokumentation wird anschaulich mit Beispielen dargelegt, wie gute BASIC-Programmierung vor sich geht. Fünf komplett beschriebene, lauffertige Anwendungsprogramme für den C-64 illustrieren den Inhalt der einzelnen Kapitel beispielhaft. Mit 64 FÜR PROFIS lernen Sie gute und erfolgreiche BASIC-Programmierung.
64 FÜR PROFIS, 1983, ca. 220 Seiten, DM 49,–.

Darauf haben Sie gewartet: Endlich ein Buch, das Ihnen ausführlich und verständlich die Arbeit mit der Floppy VC-1541 erklärt. DAS GROSSE FLOPPY BUCH ist für Anfänger, Fortgeschrittene und Profis gleichermaßen interessant. Sein Inhalt reicht von der Programmspeicherung bis zum DOS-Zugriff, von der sequentiellen Datenspeicherung bis zum Direktzugriff, von der technischen Beschreibung bis zum ausführlich dokumentierten DOS Listing, von den Systembefehlen bis zur detaillierten Beschreibung der Programme der Test/Demodiskette. Exakt beschriebene Beispiel- und Hilfsprogramme ergänzen dieses neue Superbuch. Mit dem GROSSEN FLOPPY-BUCH meistern Sie auch Ihre Floppy. DAS GROSSE FLOPPY BUCH, 1983, ca. 320 Seiten, DM 49,–.

neu **Die Diskette zum Buch mit allen wichtigen Programmen DM 39,-**



Die überarbeitete und erweiterte 2. Auflage von VC-20 TIPS & TRICKS enthält eine detaillierte Beschreibung der Programmierung von Sound und Graphik des VC-20, mehr über Speicherbelegung, Speichererweiterung und die optimale Nutzung der einzelnen Speichermodule, BASIC-Erweiterungen zum Eintippen, umfangreiche Sammlung von Poke's und anderen nützlichen Routinen, zahlreiche interessante Beispiel- und Anwendungsprogramme, komplett dokumentiert und fertig zum Eintippen (z.B. Spiele, Funktionenplotter, Graphik Editor, Sound Editor) und vieles andere mehr. VC-20 TIPS & TRICKS ist eine echte Fundgrube für jeden VC-20 Anwender.
VC-20 TIPS & TRICKS, 2. Auflage 1983, ca. 230 Seiten, DM 49,–.

Eine leicht verständliche Einführung in die Programmierung des COMMODORE 64 in Maschinensprache und Assembler für alle diejenigen, denen die Programmierung in BASIC nicht mehr ausreicht. Beispiele erläutern jeden neuen Befehl. Zur komfortablen Eingabe und zum Austesten Ihrer Maschinenprogramme enthält das Buch einen kompletten Assembler, einen Disassembler und einen Einzelschritt-Simulator, der besonders für den Anfänger sehr nützlich ist. Natürlich zugeschnitten auf Ihren Computer, den COMMODORE 64. DAS MASCHINENSPRACHEBUCH ZUM COMMODORE 64, 1984, ca. 200 Seiten, DM 39,–.

neu **Die Diskette zum Buch mit allen wichtigen Programmen DM 39,-**

IHR GROSSER PARTNER FÜR KLEINE COMPUTER

DATA BECKER

Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010 · im Hause AUTO BECKER

DATA BECKER BÜCHER und PROGRAMME erhalten Sie im Computer-Fachhandel, in den Computerabteilungen der Kauf- und Warenhäuser und im Buchhandel.
Auslieferung für Österreich Fachbuchcenter ERB, Schweiz THALI AG und Benelux COMPUTERCOLLECTIEF.

BESTELL-COUPON

Einsenden an: DATA BECKER · Merowingerstr. 30 · 4000 Düsseldorf 1
Bitte senden Sie mir:

zzgl. DM 5,– Versandkosten
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei
☐ DATA WELT 1/84 (DM 4,– in Briefmarken liegen bei)

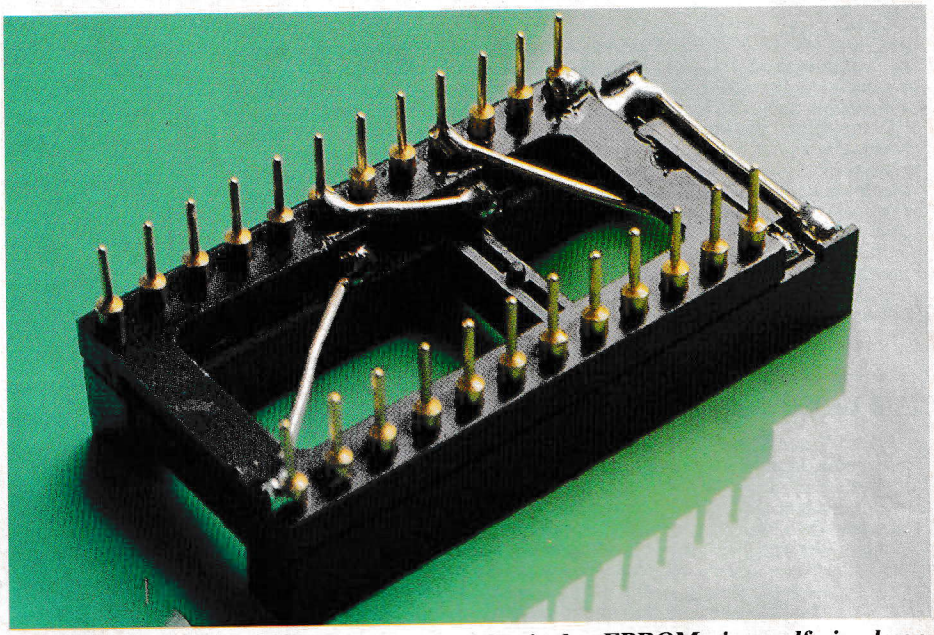
Name und Adresse bitte deutlich schreiben

EPROM statt ROM

Ein geändertes Betriebssystem
bringt viele Vorteile für 64er Besitzer.
Aber es sollte auf einer soliden Basis stehen...

In der letzten Ausgabe haben wir uns Gedanken über den Einbau eines IEC-Bus-Interfaces in unseren C64 gemacht; dazu wollen wir hier einen ersten Grundstein legen.

Leider ist der Commodore 64 nicht so perfekt geraten, wie man nach der Wahl zum Homecomputer des Jahres 1983 eigentlich vermutet. So haben sich sogar in sein Innerstes, das Betriebssystem, Fehler eingeschlichen. Wer schon einmal versucht hat, über die RS232-Schnittstelle mit anderen Rechnern zu kommunizieren, kann davon ein Lied singen. Er wird nämlich festgestellt haben, daß es mit dem 'Händeschütteln', dem sogenannten 'Handshake', nicht so richtig klappt. Unangenehm fällt der C 64 auch dann auf, wenn er mit einem PRINT FRE(0) nach seinem freien Speicherplatz gefragt wird. Er meldet sich nämlich stets mit einer negativen Zahl. Den korrekten Wert erhält man erst nach Abzug der negativen Zahl von 65 536 Byte. Um derartige Fehler im BASIC- oder Betriebssystem-ROM mittels eines selbstgebrannten EPROMs beseitigen zu können, bedarf es einiger Übung, die wir beim Leser von COMPUTER Praxis nicht voraussetzen wollen. Darum fangen wir ganz klein an.

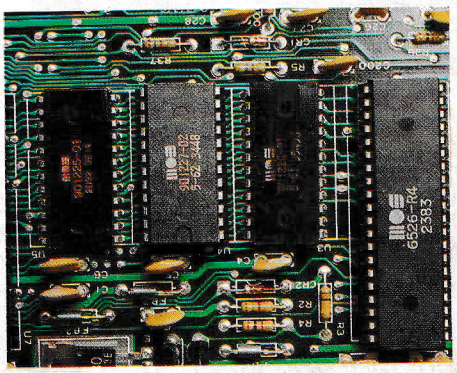


So muß der IC-Sockel verkabel werden, damit das EPROM einwandfrei gelesen werden kann.

Vom ROM ins RAM

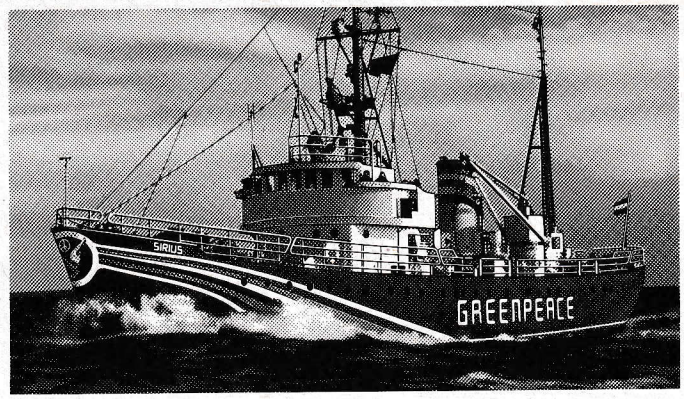
Zuallererst müssen wir wissen, in welchem Speicherbereich BASIC und Betriebssystem anzutreffen sind. An dieser Stelle soll die Angabe des Bereiches genügen (BASIC: $A000-BFFF und Betriebssystem: $E000-FFFF), wer mehr wissen will, der sei auf das Data Becker Buch "64 Intern" verwiesen, in dem sich ein kommentiertes Listing des Betriebssystems befindet. Diese beiden Speicherbereiche sind aber, und das ist ein großer Vorteil des 64er, zweimal vorhanden: einmal als ROM und parallel dazu als RAM. Wir bekommen dadurch die Möglichkeit, das vorhandene ROM in den RAM-Bereich zu übertragen, dort Veränderungen vornehmen zu können und zu testen, bevor wir ein EPROM brennen. Wichtig ist dabei, nach dem Kopiervorgang auf den RAM-Speicher umzuschalten. Das wird durch ein POKE 1,53 erreicht; ein POKE 1,55 schaltet die beiden ROMs wieder ein. Soll nur das BASIC ausgeschaltet werden, so genügt ein POKE 1,54, d.h., wir setzen lediglich Bit 0 dieser Speicherstelle auf Null.

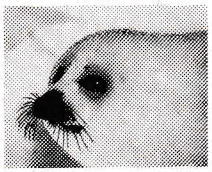
Versucht doch einmal, eigene Routinen anstelle unwichtiger zu entwerfen, die ihr in das BASIC oder das Betriebssystem einbaut. Möglichkeiten gibt es genug: Eine CATALOG-Funktion, LOAD"xxx" anstelle von LOAD"xxx", 8, andere Bildschirmfarben, eine Repeat-Funktion, automatisches Abfragen des Floppyfehlerkanals, veränderte oder neue BASIC-Befehle, Einbau eines Maschinensprachemonitors, etc.. Herauswerfen könnt ihr evtl. die RS232-Routine oder die Kassettentreiber-Software. Auf eure Ergebnisse warten wir voll höchster Spannung.



Das helle IC muß durch das EPROM ersetzt werden.

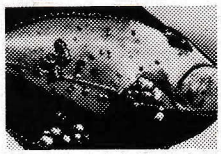
Um die Ergebnisse 'nichtflüchtig' zu machen, müßten wir die vorhandenen 8K-ROMs, sie sind zum Glück schon gesockelt, durch selbstprogrammierte EPROMs des Typs 2764 ersetzen. Leider hat dieses EPROM 28 Pins, das Rom aber nur 24. Zur Lösung des Pro-





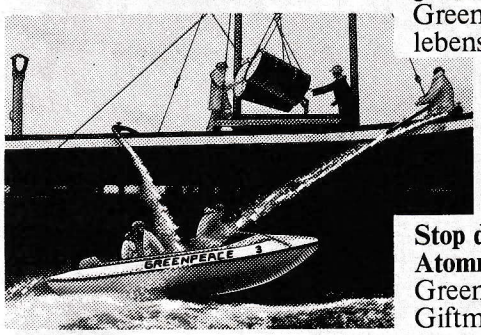
O & MD Hamburg

Schicken Sie uns den Coupon, und Sie erfahren, was Sie gemeinsam mit Greenpeace zur Rettung unserer Umwelt tun können.



Die Vergiftung der Meere fordert ihre Opfer.

Jeden Tag stirbt eine Tierart – jede Stunde eine Pflanzenart. Für die Natur ist es fünf vor zwölf. Darum müssen wir jetzt handeln. Jeder für sich und gemeinsam mit Greenpeace. Gemeinsam mit Greenpeace können wir die lebensnotwendigen Veränderungen erreichen, die wir für eine Zukunft in einer gesunden Welt brauchen. Weltweit arbeitet Greenpeace an Zielen, die für uns alle lebenswichtig sind:



Greenpeacer in Schlauchbooten behinderten die Versenkung von Atommüll im Atlantik.

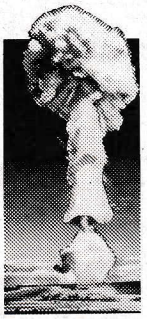
Beendigung aller Atomwaffentests.
Greenpeacer verhinderten Atombombenexplosionen in Alaska, im Pazifik und in der Wüste von Nevada, sie demonstrierten in den USA und in der Sowjetunion.

Stop der Versenkung von Chemie- und Atommüll im Meer.
Greenpeacer blockierten das Auslaufen von Giftmüllschiffen, behinderten die Verklappung auf hoher See. Sie ketteten sich auf Atommüllfrachtern an. Sie steuerten ihre Schlauchboote unter die Abwurfkräne der Atommüllfrachter.

Für die Erhaltung der letzten Robben, Wale und Delphine.
Greenpeacer manövrierten sich in Schlauchbooten zwischen Wale und Harpunenkanonen, starteten weltweite Aktionen zur Rettung der Robben, befreiten Delphine aus den Netzen japanischer Fischer.

Juni 1982: 200 Luftballons mit Karten, die den Stop sowjetischer Atomwaffentests forderten, stiegen von Bord der Sirius in den Himmel über Leningrad.

Greenpeacer besetzten den Schornstein der Hamburger Chemiefabrik Boehringer, um gegen die Herstellung hochgiftiger Insekten- und Unkrautvernichtungsmittel zu demonstrieren.



Jede Woche detoniert eine Atombombe – als Test.

Stop dem Sauren Regen und der weiteren Vergiftung unserer Umwelt durch Schadstoffe und Abgase.
Greenpeacer besetzten Schornsteine und brachten den Skandal um den Verbleib der Seveso-Giftmüllfässer ins Rollen.
Doch nicht nur spektakuläre Greenpeace-Aktionen weisen auf Umweltprobleme hin: Greenpeacer sammeln Unterschriften für Eingaben an Ministerien, sie leisten Aufklärungsarbeit, sie setzen Behörden unter Druck, damit Gesetze und internationale Abkommen zum Schutz unserer Umwelt verbessert und vor allen Dingen eingehalten werden.
Greenpeace braucht Ihre Hilfe und Unterstützung. Machen Sie mit, und helfen Sie mit. Informieren Sie sich, wie Sie bei Greenpeace mitarbeiten können. Schicken Sie den ausgefüllten Coupon heute noch ein.

GREENPEACE

Greenpeace e. V.
Hohe Brücke 1 – Haus der Seefahrt –
2000 Hamburg 11
Telefon: 040/37 33 44 oder 37 33 59.

Ich will dazu beitragen, unsere Umwelt zu erhalten. Ich will wissen, wie ich das gemeinsam mit Greenpeace erreichen kann. Bitte senden Sie mir ausführliches Informationsmaterial.

Name/Vorname

Straße/Nummer

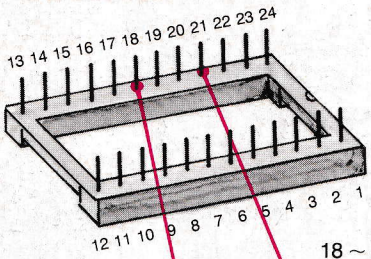
PLZ/Wohnort A 187

Sie zu informieren kostet uns DM 2,40. Schicken Sie diesen Betrag in Briefmarken (falls mehr, bitte Scheck) zusammen mit diesem Coupon an:
GREENPEACE-Informationsdienst
Hohe Brücke 1 – Haus der Seefahrt –
2000 Hamburg 11

Sockel-Konvertierung Kernal Rom → 2764

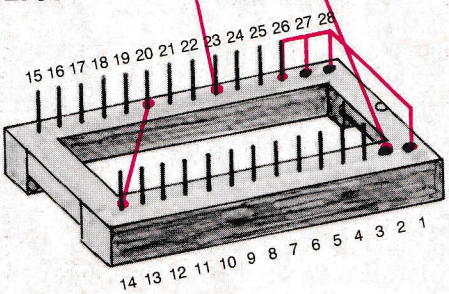
Pin 3 (2764) ~ Pin 1 (Rom)

Rom-Sockel



18 ~ A 11
21 ~ A 12
(ähnlich wie 2532)

2764



20 ~ $\overline{CE}$
2 ~ A 12
23 ~ A 11

Am Eprom Pin 20 und 23 wegbiegen; Pin 1, 2, 27 und 28 stecken nicht im Rom-Sockel.

blems müssen wir uns einen »Konvertierungssockel« bauen. Einen solchen Sockel seht ihr auf unserer Abbildung. Er paßt genau in den Sockel des C 64 und kann den 2764 aufnehmen. Wie ihr sehen könnt, trägt dazu ein 24poliger einen 28poligen Sockel 'Huckepacke'. Da vier Pins zuviel an Bord sind, müssen die Pins 1,2,27 und 28 des oberen am unteren Sockel 'vorbeigestreckt' und zwei weitere Beinchen nach innen geknickt und durch zwei angelötete Drähte vertauscht werden. Der Rest paßt zum Glück genau zu- oder besser aufeinander. Die genaue Verbindung könnt ihr der obenstehenden Zeichnung entnehmen, und es bleibt uns nicht mehr, als euch recht viel Spaß beim Basteln zu wünschen. ■

J.-P.P./Bernd Schwermer

WER ENTFÜHRT EUCH IN DIE WUNDERWELT DER WISSENSCHAFT?

COMMODORE COMPUTER.

Den einen führt der Commodore-Heimcomputer von den ersten Schritten der Physik in die grenzenlose Welt der Astrophysik. Den anderen von Bio und Chemie in die irdische Welt der Biochemie.

Ein faszinierendes Ding: ein echter Computer mit unbegrenzten Möglichkeiten. Mit ihm kann man spielend die Weltsprachen der Computer lernen. Kann man Daten, Adressen oder Plattensammlungen organisieren. Sogar videospielen kann man damit.

Ein tolles Ding: ein echter Computer für eine gute Idee nach der anderen. Der Commodore-Heimcomputer. Er kostet nicht die Welt.

Beim Commodore-Vertragshandel, in führenden Warenhäusern, guten Rundfunk- und Fernsehfachgeschäften und großen Versandhäusern.

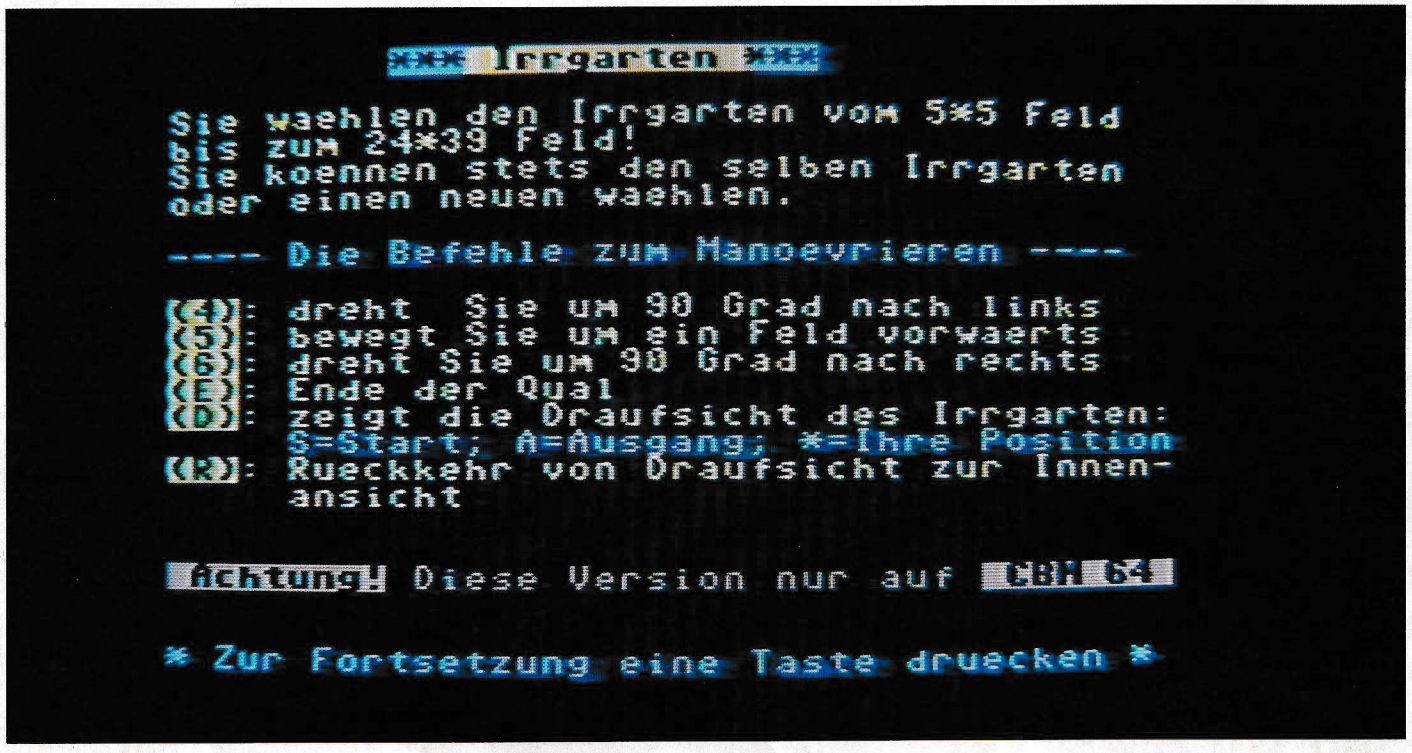
Mehr Informationen gibt's von: Commodore Büromaschinen GmbH, Abt. MK, Lyoner Straße 38, 6000 Frankfurt 71. Die Anschrift des Commodore-Fachhändlers in Ihrer Nähe erfahren Sie telefonisch von den Commodore-Verkaufsbüros: Düsseldorf 02 11/31 20 47/48, Frankfurt 06 11/6 63 81 99, Hamburg 0 40/21 13 86, München 0 89/46 30 09, Stuttgart 07 11/24 73 29, Basel 0 61/23 78 00, Wien 02 22/67 56 00.

Commodore

Eine gute Idee nach der anderen.

3-D-LABYRINTH

Die geheimen Winkel des CBM 64



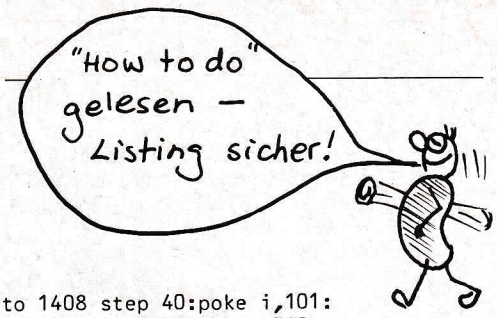
Schon wieder ein Irrgartenspiel — werden jetzt viele von euch sagen, womit sie ja auch nicht ganz unrecht haben. Das 3-D-Labyrinth zeichnet sich aber dadurch aus, daß es dem C 64 User, der sich noch nicht mit den Graphikmöglichkeiten seines Rechners befaßt hat, einen Einstieg ermöglicht. Für schon etwas erfahrenere Benutzer bietet sich aber auch noch an, das Programm auszubauen. Dies könnte durch Sound und Multicolor-Graphik realisiert werden. Wem auch das noch nicht ausreicht, der könnte ein nettes Adventure daraus machen.

Hier ist also zunächst mal das Labyrinth in dreidimensionaler Darstellung. Das Ziel ist es, in möglichst kurzer Zeit den Ausgang des Irrgartens zu finden. Am Anfang des Spiels hat man die Option, die Größe des Irrgartens zu wählen und zwar vom kleinsten mit 5 mal 5 Räumen bis zum größten mit 24 mal 39 Räumen. Diese Beschränkung nach oben ergibt sich einfach aus der möglichen Bildschirmdarstellung (25 Zeilen, 40 Spalten) der Draufsicht unseres irren Irrgartens.

Die Bedienungsanleitung steht im Programm in den Zeilen 1530—1690. Die Zifferntasten 4, 5, 6 dienen der Bewegung (links, geradeaus, rechts), die Buchstabentasten E, D, R diversen Optionen (Ende, Draufsicht, Rückkehr von Draufsicht zu Innenansicht). Die anderen Teile des Programms sind durch entsprechende REM-Zeilen gut nachvollziehbar strukturiert:

Zeilen	Erklärung
100	Rahmen und Hintergrund werden 'schwarzgepoket'. Sprung in die Anleitung.
102— 570	Als Unterprogramme sind hier die Graphiken für die Innenansicht.
572— 720	ruft die Unterprogramme der Innen-Graphik in die Zeilen 102—570 auf.
725— 840	fragt per GET die Tastatur ab und setzt die Variablen der Position und Bewegungsrichtung entsprechend.
850—1030	Diese Zeilen werden aufgerufen, wenn der Ausgang erreicht worden ist, eine Graphik wird aufgebaut und die benötigte Zeit schließlich ausgegeben.
1050—1210	Der Irrgarten wird konstruiert.
1220—1290	Die Draufsicht wird erzeugt.
1292—1330	In diesen Zeilen wird die Softwareuhr (TI$) abgefragt und ausgegeben.
1530	In Zeile 1530 wird der Graphikmodus ausgeschaltet und auf Schriftausgabe umgeschaltet.
1540—1680	Das Spiel und die Tastenbelegung werden erklärt.
1690—1820	In diesen Zeilen muß der Spieler die Eingaben für den Irrgarten machen.
1890	Die Konstruktionsroutine wird aufgerufen, die Uhr auf Null gesetzt, der Graphikmode wieder auf Graphik umgeschaltet und das Spiel gestartet.

Die Zeilen 102—507 sind wegen der zahlreichen Pokebefehle mit besonders großer Sorgfalt einzugeben. Aber dann: Viel Spaß im Irrgarten.




```
100 poke 53281,0:poke 53280,0:goto  1520
102 rem  *****************
104 rem  begrenzungslinien
106 rem  *****************
110 for i=1028 to 1868 step 40:poke i,103:poke fb+i,7:
    next :if  j<>3 goto  130
120 for i=1908 to 1986 step 39:poke i,78:poke fb+i,7:
    next :return
130 for i=1908 to 1905 step -1:poke i,99:poke fb+i,7:
    next  :return
140 for i=1060 to 1900 step 40:poke i,101:
    poke 54272+i,7:next :if  j<>3 goto  160
150 for i=1940 to 2022 step 41:poke i,77:poke fb+i,7:
    next :return
160 for i=1940 to 1943:poke i,99:poke fb+i,7:next :
    return
170 for i=1033 to 1673 step 40:poke i,103:poke fb+i,7:
    next :if  j<>3 goto  190
180 for i=1713 to 1869 step 39:poke i,78:poke fb+i,7:
    next :return
190 for i=1713 to 1709 step -1:poke i,99:poke fb+i,7:
    next :return
200 for i=1055 to 1695 step 40:poke i,101:poke fb+i,7:
    next :if  j<>3 goto  220
210 for i=1735 to 1899 step 41:poke i,77:poke fb+i,7:
    next :return
220 for i=1735 to 1739:poke i,99:poke fb+i,7:next :
    return
230 for i=1037 to 1517 step 40:poke i,103:poke fb+i,7:
    next :if  j<>3 goto  250
240 for i=1557 to 1674 step 39:poke i,78:poke fb+i,7:
    next :return
250 for i=1557 to 1554 step -1:poke i,99:poke fb+i,7:
    next :return
260 for i=1051 to 1531 step 40:poke i,101:poke fb+i,7:
    next :if  j<>3 goto  280
270 for i=1571 to 1694 step 41:poke i,77:poke fb+i,7:
    next :return
280 for i=1571 to 1574:poke i,99:poke fb+i,7:next :
    return
290 for i=1160 to 1400 step 40:poke i,103:poke fb+i,7:
    next :if  j<>3 goto  320
300 poke 1440,78:poke 1479,78:poke 1518,78:
    poke 55712,7:poke 55751,7:poke 55790,7
310 for i=1038 to 1120 step 41:poke i,77:poke i+fb,7:
    next :return
320 for i=1120 to 1118 step -1:poke i,100:
    poke i+320,99:poke i+fb,7:poke 54592+i,7
325 next :return
330 for i=1168 to 1408 step 40:poke i,101:
    poke fb+i,7:next :if  j<>3 goto  360
340 poke 1448,77:poke 1489,77:poke 1530,77:
    poke 55720,7:poke 55761,7:poke 55802,7
350 poke 1128,78:poke 1089,78:poke 1050,78:
    poke 55400,7:poke 55361,7:poke 55322,7
355 return
360 for i=1128 to 1130:poke i,100:poke i+320,99:
    poke fb+i,7:poke 54592+i,7:next
365 return
370 poke 1242,103:poke 1282,103:poke 1322,103:
    poke 55514,7:poke 55554,7
375 poke 55594,7:if j<>3 then 390
380 poke 1161,77:poke 1202,77:poke 1401,78:
    poke 1362,78:poke 55433,7:poke 55474,7
385 poke 55673,7:poke 55634,7:return
390 poke 1201,100:poke 1202,100:poke 1203,99:
    poke 1204,99
395 for i=1 to 4:poke 55472+i,7:next :return
400 poke 1246,101:poke 1286,101:poke 1326,101:
    poke 55518,7:poke 55558,7:poke 55598,7
405 if j<>3goto 420
410 poke 1206,78:poke 1167,78:poke 1366,77:
    poke 1407,77:poke 55478,7:poke 55439,7
415 poke 55638,7:poke 55679,7:return
420 poke 1206,100:poke 1207,100:poke 1366,99:
    poke 1367,99
425 poke 55478,7:poke 55479,7:poke 55638,7:
    poke 55639,7:return
430 poke 1283,103:poke 55555,7:if  j<>3 goto  450
440 poke 1243,77:poke 1323,78:poke 55515,7:
    poke 55595,7:return
450 poke 1243,100:poke 1323,99:poke 55515,7:
    poke 55595,7:return
460 poke 1285,101:poke 55557,7:if  j<>3 goto  480
470 poke 1245,78:poke 1325,77:poke 55517,7:
    poke 55597,7:return
480 poke 1245,100:poke 1325,99:poke 55517,7:
    poke 55597,7:return
490 for i=1909 to 1939:poke i,99:poke fb+i,7:next :
    return
500 for i=1714 to 1734:poke i,99:poke fb+i,7:next :
    return
510 for i=1558 to 1570:poke i,99:poke fb+i,7:next :
    return
520 for i=1441 to 1447:poke i,99:poke i-320,100:
    poke fb+i,7:poke 53952+i,7:next
525 return
530 for i=1363 to 1365:poke i,99:poke i-160,100:
```


McGraw-Hill Anwenderhandbücher

448 Seiten, DM 39,80

- Komplette Beschreibung von System und Zusatzgeräten
- Programmier-Technik
- Colorgrafik
- Tonerzeugung
- Nachschlagewerk

388 Seiten, DM 32,-

Fordern Sie unseren Gesamtprospekt an! - Coupon ausschneiden und einsenden an:
McGraw-Hill
Book Co. GmbH
Lademannbogen 136
2000 Hamburg 63

Bitte senden Sie mir den Gesamtprospekt

Name

Anschrift

```
    poke fb+i,7:poke 54112+i,7:next
535 return
540 poke 1324,99:poke 1244,100:poke 55596,7:
    poke 55516,7:return
550 poke 1284,86:poke 55556,7:return
560 poke 1284,147:poke 55556,7:return
570 poke 1284,129:poke 55556,7:return
572 rem  ***************
574 rem  zeichenroutine
576 rem  ***************
580 ct=0
590 j=m((ai+di*ct-dj),(aj+dj*ct+di))
600 on  ct+1 gosub  110,170,230,290,370,430,460
610 j=m(ai+di*ct+dj,aj+dj*ct-di)
620 on  ct+1 gosub  140,200,260,330,400,460
630 ct=ct+1:j=m(ai+di*ct,aj+dj*ct)
640 if (j=1) or (j=2)goto  670
650 if (j=0) or (j=3) or (j=4)goto  690
660 goto  1720
670 if  ct<6 goto  590
680 gosub  550:return
690 on  ct gosub  490,500,510,520,530,540
700 if  j=0  and  ct<1  then  560
710 if  j=4  and  ct<=3  then  570
720 return
725 rem  ****************
726 rem  tastaturabfrage
727 rem  ****************
730 get  i$:gosub  1300:if  i$=""goto  730
740 if  i$="4" then  j=dj:dj=di:di=-j:goto  830
750 if  i$="6" then  j=dj:dj=-di:di=j:goto  830
760 if  i$="5" then  ai=ai+di:aj=aj+dj:goto  810
770 if  i$="d" then  gosub  1220
780 if  i$="e" then  print "{CLR}":goto  1720
790 if  i$="r" then  830
800 goto  730
810 if  (m(ai,aj)<>3) and (m(ai,aj)<>0)goto  830
820 ai=ai-di:aj=aj-dj:goto  730
830 if  m(ai,aj)=4 goto  860
840 print "{CLR}";:gosub  580:goto  730
850 rem  ******************
852 rem  ausgang erreicht
854 rem  ******************
860 print "{CLR}";
870 for  c=1  to  40  step  10
880 print "{HOME}{BLU}";
890 print tab(c)"{CHR$(172)}{CHR$(187)}   "
900 print tab(c)"{CHR$(190)}{RVS ON}{CHR$(162)}
    {RVS OFF}{CHR$(188)}{RVS ON}{RVS OFF}  "
910 print tab(c)"{CHR$(188)}{RVS ON} F{CHR$(190)}
    {RVS OFF} "
920 print tab(c)"{CHR$(162)}{RVS ON}{CHR$(162)}
    {RVS OFF}   "
930 print tab(c)"{CHR$(191,161)}{RVS ON}
    {CHR$(161,191)}{RVS OFF} "
940 print tab(c)"{CHR$(188)}{CHR$(190)}{RVS ON}
    {CHR$(188)}{CHR$(190)}{RVS OFF} "
950 print tab(c)"{CHR$(161)}{RVS ON}{RVS OFF}
    {CHR$(161)}   "
960 print tab(c)"{CHR$(172)}{CHR$(190)}{RVS ON}
    {RVS OFF}{CHR$(188)}{RVS ON}{CHR$(187)}{RVS OFF} ";
970 next
980 print :print :print :print
990 print "{YEL}geschafft in der zeit von :"
1000 print "{CHR$(154)}"mid$(ti$,3,2)"{YEL} minuten und
    {CHR$(154)} "right$(ti$,2)"{YEL} sekunden"
1010 pk=int(mi*mj*120/val(ti$))
1020 print "damit haben sie{CHR$(154)}{RVS ON} "
    pk"{LEFT}{RVS ON}{RVS OFF}{YEL}  punkte erreicht"
1030 print :goto  1720
1050 rem  ************
1060 rem  konstruktion
1070 rem  ************
1090 cm=mi*mj/4:for  i=0  to  mi:for  j=0  to  mj:
    m(i,j)=3:next  j,i
1100 di=-1:dj=0:ai=mi:aj=fnr((mj/2)-1)*2:ct=1
1110 m(ai,aj)=0:ai=ai+di:aj=aj+dj:si=ai:sj=aj
1120 m(ai,aj)=1
1130 di=fnr(3)-2:dj=fnr(3)-2:if abs(di)=abs(dj)
    goto 1130
1140 ai=ai+di:aj=aj+dj
1150 if ai=o or ai=mi or aj=0 or aj=mj goto 1200
1160 if m(ai,aj)=1 goto 1130
1170 for i=-1 to 1 step 2:for j=-1 to 1 step 2
1180 if m(ai+i,aj+j)=1 and m(ai,aj+j)=1 and
    m(ai+i,aj)=1 goto 1210
1190 next j,i:ct=ct+1:goto 1120
1200 if ct>cm then m(ai,aj)=4:ai=si:aj=sj:return
1210 ai=ai+di:aj=aj-dj:goto 1130
1220 print "{CLR}";:for i=0 to mi:for j=0 to mj
1230 if m(i,j)=1 or m(i,j)=2 then k=32
1240 if m(i,j)=3 then k=76
1250 if m(i,j)=0 then k=19
1260 if m(i,j)=4 then k=1
1270 if ai=i and aj=j then k=42
1280 poke 1065+40*(i-1)+j-1,k:poke 55337+40*(i-1)+j-1,7
1290 next  j,i:return
1292 rem  ***********
1294 rem  zeitzaehler
1296 rem  ***********
1300 i=2006:poke fb+i,14:poke fb-1+i,14:poke fb-3+i,14:
    poke fb-4+i,14
1310 poke  i,48+val(mid$(ti$,6,1)):
    poke  i-1,48+val(mid$(ti$,5,1)):poke  i-2,136
1320 poke  i-3,48+val(mid$(ti$,4,1)):
    poke  i-4,48+val(mid$(ti$,3,1))
1330 t=val(ti$):return
1510 rem  ***************
1520 rem  spielanleitung
1525 rem  ***************
1530 poke  53272,23
1540 print "{CLR}";tab(9);"{CHR$(154)}{RVS ON}***{YEL}
    Irrgarten{CHR$(154)} ***{RVS OFF}"
1550 print :print "{YEL}Sie waehlen den Irrgarten vom
    5*5 Feld  bis zum 24*39 Feld!"
1560 print "Sie koennen stets den selben Irrgarten
    oder einen neuen waehlen."
1570 print :print "{CHR$(154)}
    ---- Die Befehle zum Manoevrieren ----":print
1580 print "{RVS ON}{YEL}(4){RVS OFF}
    : dreht  Sie um 90 Grad nach links"
1590 print "{RVS ON}(5){RVS OFF}
    : bewegt Sie um ein Feld vorwaerts"
1600 print "{RVS ON}(6){RVS OFF}
    : dreht Sie um 90 Grad nach rechts"
1610 print "{RVS ON}(E){RVS OFF}
    : Ende der Qual"
1620 print "{RVS ON}(D){RVS OFF}
    : zeigt die Draufsicht des Irrgarten:"
1630 print "{UP}{CHR$(154)}{5 SPACE}
    S=Start; A=Ausgang; *=Ihre Position"
1640 print "{UP}{RVS ON}{YEL}(R){RVS OFF}
    : Rueckkehr von Draufsicht zur Innen-{5 SPACE}
    ansicht":print
1650 print "{DOWN}{CHR$(150)}{RVS ON} Achtung!
    {RVS OFF} Diese Version nur auf{RVS ON}  CBM 64
    {RVS OFF} "
1680 print :print "{CHR$(154)}
    * Zur Fortsetzung eine Taste druecken *"
1690 get  i$:if  i$="" goto  1690
1710 fb=54272:dim  m(24,39):def  fnr(k)=int(k*rnd(1))+1:
    print "{CLR}";:goto  1800
1720 print "{YEL}wollen  sie  nochmal  den  gleichen"
1725 print "irrgarten oder einen anderen ? (g/a)"
1730 get  i$:if  i$="" then  1730
1740 if  i$="g" then  ai=si:aj=sj:ti$="000000":t=0:
    goto 840
1750 if  i$<>"a" goto  1730
1770 print "{CLR}"
1800 poke 53272,23:print "{CHR$(154)}
     *  Jetzt  muessen  Sie  nur  noch die "
1801 print "{DOWN} *{RVS ON}  Groesse{RVS OFF}
     des Irrgartens festlegen."
1802 print "{DOWN} *  Bitte  geben  Sie  mir  die"
1805 input "{DOWN} *{YEL}  Laenge und Breite ";mi,mj
1810 if (mi<5) or (mi>24) or (mj<5) or (mj>39) then
     print "{DOWN}{RVS ON} Halt!{RVS OFF}
     5*5 bis 24*39{DOWN}":goto 1800
1820 print "{3 DOWN}{YEL}  Ich konstruiere ---
    {CHR$(154)}{RVS ON}  bitte warten!{RVS OFF}"
1830 gosub  1090:ti$="000000":t=0:poke  53272,21:
    goto  840 ■
```

Michael Schaffer

Der „neue"
DATA BECKER

Das müssen Sie gesehen haben!

Der „neue" DATA BECKER. Umgebaut, neugestaltet und erweitert. Ein Computer-Kaufhaus wie Sie es schon immer gesucht haben:

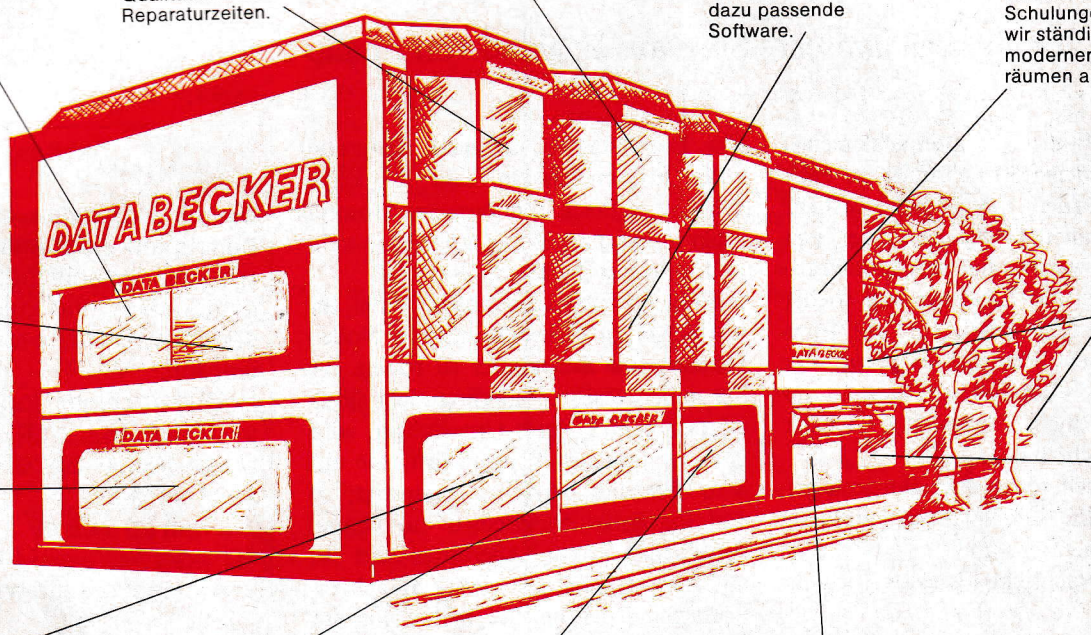
Hier überprüfen unsere Software- und Hardwareexperten für Sie das internationale Angebot und wählen neue Hits für unser Sortiment aus.

Das ist wichtig: was wir verkaufen reparieren wir auch. Unser erfahrenes Technik-Team garantiert für hohe Qualität und kurze Reparaturzeiten.

DATA EXPRESS heißt unser neuer Versandservice, der Ihre Bestellungen schnell und zuverlässig ausführt.

Hier finden Sie nicht nur eine große Auswahl an Bürocomputern, z.B. von Apple, Commodore, EPSON, IBM und Sirius, sondern auch die dazu passende Software.

Sie wollen mehr wissen und dazulernen? Dann nehmen Sie doch an unseren Seminaren und Schulungen teil, die wir ständig in unseren modernen Schulungsräumen abhalten.



Hier steht in unserer großräumigen Systemausstellung der neue Apple MC-Intosh, den Ihnen unsere geschulten Systemberater gerne zeigen.

Hier geht's zu AUTO BECKER, dem interessantesten Autoladen der Welt, der sich im gleichen Haus befindet.

Hier finden Sie nicht nur die bekannten DATA BECKER BÜCHER, sondern über 1000 (!) verschiedene Buchtitel, darunter viele Spezialbücher zu einzelnen Mikrocomputern.

Hier hat unser Verkaufsleiter Platz freigelassen für einen neuen Superhit von Apple, den wir im Mai erwarten.

Wem die Glotze nicht reicht, der findet bei uns eine große Monitorauswahl vom preiswerten 9" Gerät bis zum großen Farbmonitor.

Auch für Computerbesitzer ist der beste Platz an der Theke, nämlich an der DATA BECKER Softwaretheke, wo Ihnen unsere freundlichen Experten gerne aus unserer riesigen Auswahl das richtige Programm empfehlen.

An unserer Druckerwand finden Sie eine große Palette von Druckern für jeden Zweck und Geldbeutel, natürlich ansprechend erklärt.

Natürlich finden Sie bei uns ständig aktuelle Sonderangebote und preiswerte Gebrauchtgeräte.

Was Sie hier nicht sehen können, ist unser neues Zentrallager, in dem z.B. über 100.000 Commodore 64 Platz fänden. So sind wir immer gut lieferfähig.

Über 1000 qm Ausstellungsfläche. Über 20 freundliche Verkaufsberater. Umfassende Auswahl, qualifizierte Beratung, attraktive Preise und ein zuverlässiger Service.

IHR GROSSER PARTNER FÜR KLEINE COMPUTER

DATA BECKER

Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010 · im Hause AUTO BECKER

INFO – COUPON

Bitte schicken Sie mir kostenlos und unverbindlich Ihr Informationspaket mit Hausprospekt, Lageplan, Versandangebot und Seminarterminen.

Name und Adresse bitte deutlich schreiben

BRICK UP

Ein intergalaktisches Spiel aus England von Phillip Smith für den VC 20

Als Mitglied der intergalaktischen Friedensflotte befindet ihr euch auf einem eurer Routinebesuche auf dem Planeten Tarxxon. Wie nicht anders zu erwarten, hat sich euer Bordcomputer bei der Landung mal wieder verkalkuliert und ihr seid statt auf Tarxxon auf einem Planeten am Ende der Milchstraße heruntergegangen, der nichts als Steine und böswillige Monster hervorgebracht hat.

Wieder meltet sich euer Bordcomputer und kündigt euch eine gute und eine schlechte Nachricht an:

Die gute Nachricht: Es nähert sich ein Feind, und wo viel Feind, da ist viel Ehr. Die schlechte Nachricht: Aufgrund eurer Friedensmission habt ihr keine Waffen an Bord. Ein garstiges Monster rückt immer nächer und eine Frage stellt sich immer dringlicher: Was tun?

Es bleibt euch keine andere Wahl, als das Schiff zu verlassen und mit bloßen Händen eine Steinmauer zu bauen, die der außerirdische Widerling nicht überwinden kann. Dieses Unterfangen wird von der geringen Schwerkraft, die auf dem Planeten herrscht, begünstigt, so daß ihr ganze Steinreihen auf einmal verschieben könnt. Allerdings habt ihr auf euren Sauerstoffvorrat zu achten, denn ohne geht nichts mehr, und ihr

Variablen-Tabelle

FNX (Zeile 7)	bestimmt die X-Koordinate des Monsters.
FNY (Zeile 6)	bestimmt die Y-Koordinate des Monsters.
D	beinhaltet den Schwierigkeitsgrad (1—9).
P	beinhaltet die Speicherzelle im Bildschirmspeicher, wo das Zeichen des Spielers sich gerade befindet.
P1	beinhaltet die Zahl, die zu P nach der Bewegungsabfrage dazuaddiert wird, und somit die Bewegungsrichtung angibt.
K	Wert der Taste, die gerade gedrückt ist.
H	Wert der Position, in der sich der Joystick gerade befindet.
Q	Enthält im Normalfall die gleiche Zahl wie P, wird jedoch zur Überprüfung genommen, ob auf dem Feld, auf das man ziehen will, bereits ein anderes Zeichen als das Leerzeichen steht (Zeile 150—170).
TL	Zeit, nach der das Spiel beendet ist.
SC	beinhaltet den momentanen Punktestand.
HI	beinhaltet den Hiscore.

Achtung!

Dieses Programm definiert den benötigten Zeichensatz. Es muß vor dem eigentlichen Programm eingegeben und dann auf Kassette bzw. Diskette abgespeichert werden.

Bei Einsatz einer Floppy-Disk muß in Zeile 16…+",1{2*UP}{WHT}"; durch 16…+",8{2*UP}{WHT}"; ersetzt werden.

```
 0 rem (c) jens ey
10 print "{clr}{blk} {10*down}
   {5 space} bitte warten"
11 for t=0 to 512
12 poke 7168+t, peek (32768+t)
13 next t
14 for t=1 to 3: read a:
   for s=0 to 7: read b:
   poke 7168+a*8+s,b: next s,t
15 print "{6 space}{down}
   loading !!"
16 print "load"+chr$(34)+
   "brick-up II"+chr$(34)+
   ",1{2*up}{wht}";
17 poke 198,6: poke 631,144:
   poke 632,13: poke 633,82:
   poke 634,85: poke 635,78:
   poke 636,13
18 poke 55,0: poke 56,28: new
20 data 0,0,127,127,127,127,127,
   127,0
30 data 27,60,60,24,126,189,189,
   36,102
40 data 28,24,60,90,126,102,60,
   24,36
```

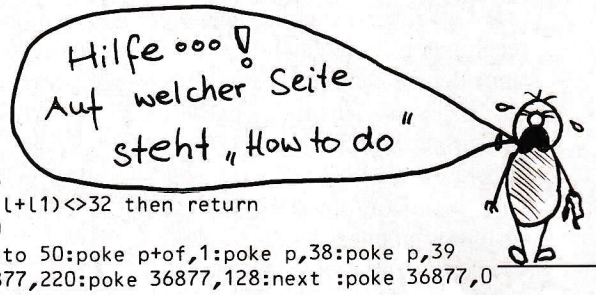
seid auf ewig verloren.

Zu spielen ist dieses Spiel wahlweise mit Joystick oder Tastatur.

Um das relativ kurze Programm für den VC 20 ohne Erweiterung zu verstehen, ist es ganz nützlich, die Variablenbelegung zu kennen. Dazu schnell eine Tabelle, die die Aufgaben der einzelnen Variablen aufführt.

Interessant an diesem Spiel ist vor allem die Abfrage der im Bildschirmspeicher stehenden Werte. Diese Methode ist sehr gebräuchlich, wenn man Spiele programmiert, bei denen es auf Schnelligkeit ankommt. Der Bildschirmspeicher beim VC 20 ohne Erweiterung liegt von 7680 bis 8185. Probiert einmal mit POKE einige Zeichen auf den Bildschirm zu bringen und so Graphiken und Spiele zu entwerfen.

Ein letzter Tip von uns: Solltet ihr vorhaben, das vorliegende BRICK UP zu verändern, so erledigt das, bevor ihr das Spiel startet. BRICK UP arbeitet nämlich mit geändertem Zeichensatz, und das macht es schwierig, nach Änderung zu programmieren.




```
  6 def fny(xy)=int((xy-7680)/22)
  7 def fnx(xy)=xy-7680-22*fny(xy)
  8 d=9:poke 37139,0:poke 37154,127
 10 poke 36869,255:poke 36879,90:poke 36878,15:
    of=30720
 15 gosub 500
 20 print CHR$(8)'{CLR} {RED} {RVS ON} {22*CHR$(218)}';
 30 for 1=0 to 18:print "{CHR$(218)} {RVS OFF} {20*SPACE}
    {RVS ON} {CHR$(218)} ";:next
 40 print "{22*CHR$(218)} {RVS OFF}"
 50 for i=0 to d*5+40
 60 r=int(rnd(1)*461)+7680:if peek(r)<>32 then 60
 70 poke r,0:next
 80 p=7703
 90 l=8118:poke l,28:poke l+of,4
 95 ti$="000000"
100 p=p+p1:poke p,27:poke p+of,6:k=peek(197):
    h=peek(37137):p1=0
101 if ti/d=int(ti/d) then gosub 200
102 if p=l then 400
105 print "{CLR} {21*DOWN} {BLU} luft {RVS ON} : "500-int
    (ti/20)" {LEFT} liter " : if ti 10000 then 400
110 if k=13 or (h and 4)=0 then p1=p1-22
120 if k=21 or (h and 16)=0 then p1=p1-1
130 if k=22 or (peek(37152) and 128)=0 then p1=p1+1
140 if k=37 or (h and 8)=0 then p1=p1+22
145 if p1=0 then 100
150 q=p
156 poke 36876,250:for i=0 to 10:next :poke 36876,0
160 z=peek(q+p1):if z=32 then 190
165 if z>27 then p1=0:goto 100
170 q=q+p1:goto 160
190 for i=q+p1 to p step -p1:poke i,peek(i-p1):next
199 poke p,32:poke p+of,2:goto 100
200 l1=sgn(fnx(p)-fnx(l))+22*(sgn(fny(p)-fny(l)))
225 z=peek(l+l1):if z<>32 and z<>27 then 250
230 poke l,32:poke l+of,2:l=l+l1:poke l,28:poke l+of,4
240 return
250 f=0:l1=0:poke 36874,128:for i=0 to 20:next :
    poke 36874,0:if rnd(1)<.5 then f=1:goto 260
251 f=0:tp=22*(int(rnd(1)*3)-1):if tp=0 then 251
252 if peek(l+tp)=32 then 299
255 if peek(l-tp)=32 then tp=-tp:goto 299
260 tp=int(rnd(1)*3)-1:if tp=0 then 260
261 if peek(l-tp)=32 then tp=-tp:goto 299
265 if peek(l+tp)=32 then 299
270 if f=1 then 251
280 if peek(l-1)=32 or peek(l+1)=32 or peek(l+22)=32 or
    peek(l-22)=32 then 299
285 if peek(l-21)=32 or peek(l+21)=32 or peek(l+23)=32
    or peek(l-23)=32 then 299
290 goto 600
299 l1=l1+tp
300 if peek(l+l1)<>32 then return
310 goto 230
400 for i=0 to 50:poke p+of,1:poke p,38:poke p,39
410 poke 36877,220:poke 36877,128:next :poke 36877,0
415 restore
420 for i=0 to 10:read a:poke 36875,a:read b:
    for j=0 to b*200:next :poke 36875,0:next :goto 660
500 print "{CLR}{RED}{6 SPACE}@@@@@@@@@@"
510 print "{6 SPACE}@{BLU}brickup{RED}@"
520 print "{6 SPACE}@@@@@@@@@@"
525 print "{BLK} {DOWN} (c) philip smith 1983"
530 print"{DOWN} {BLK} bewege dich {BLU} [ {BLK} mit
    den tasten:"
540 print "{DOWN} {BLU} {5*SPACE} {RVS ON} p'
550 print "    {RVS ON} ] {CHR$(219)} {BLU} {RVS ON} ; {RVS
    OFF} {BLK} oder stick"
560 print "{5*SPACE} {BLU} {RVS ON}."
570 print "{DOWN} {BLK} schiebe steine {RED} @ {BLK} vor
    {4*SPACE} dir her und"
580 print "{DOWN} schliesse das {PUR} {POUND} {BLK} alien
    in den mauern ein"
590 print"{DOWN} druecke {RVS ON} {BLU} f1 {RVS OFF}
    {BLK} zum starten {RVS ON} {BLU} f3 {RVS OFF} {BLK}
    fuer levelaenderung"
595 get a$:if a$="{CHR$(134)}" or
    (peek(37137) and 4)=0 then d=d-1:if d=0 then d=9
596 if (peek(37137) and 4)=0 then 596
597 print "{YEL}level"10-d"{UP}"
598 if a$<>"{CHR$(133)}" and
    not ((peek(37137) and 32)=0) then 595
599 return
600 tl=ti:for i=0 to 50:poke l+of,7:poke l,38:
    poke l,39
610 poke 36877,220:poke 36877,128:next :poke 36877,0
620 for i=128 to 240 step 10
630 for j=i to i+15:poke 36876,j:next :next :
    poke 36876,0
635 sc=100*(10-d)+500-int(tl/20)
640 print "{BLU} {CLR} {3*DOWN} {4*RIGHT} score {RVS ON}" sc
650 if sc > hs then hs = sc : print "{3*DOWN} {4*RIGHT} {WHT}
    best score !":f1 = 1 : if d = 2 then d = d - 1
655 if f1 = 1 then f1 = 0 : goto 670
660 print "{BLK} {CLR} {6*DOWN} {4*RIGHT} high score
    {RVS ON}" hs
670 print "{BLK} {2*DOWN} druecke {RVS ON} {BLU} f1 {RVS OFF}
    {BLK} zum spielen !"
680 get a$:if a$<>"{CHR$(133)}" and not ((peek(37137)
    and 32)=0) then 680
685 if (peek(37137) and 32)=0 then 685
690 goto 15
1000 data 195,3,195,2,195,1,195,2,203,2,201,1,201,2,
    195,1,195,2,191,1,195,3
```

Die Commodores unter sich

Was soll man tun, wenn selbst Familienmitglieder sich nicht verstehen?

Bevor man sich überlegt, ob und wie man Programme anderer Systeme für den Commodore gangbar machen kann, stellt sich zuersteinmal die Frage, inwieweit die Commodorecomputer untereinander kompatibel sind. Bei dieser Fragestellung spielen vor allem die Speicheradressen von Bildschirmspeicher, Farbregister sowie BASICstart eine entscheidende Rolle.

Es ist bei einigen Programmen für Commodoremaschinen üblich, Zeichen, die auf dem Bildschirm erscheinen sollen, nicht über die Cursorsteuerungstasten printen zu lassen, sondern sie, zwecks besserer Übersicht oder höherer Geschwindigkeit, direkt in den Bildschirmspeicher zu schreiben. Nachteil dieser Programmiertechnik ist allerdings, daß die so geschriebenen Programme nur noch auf Maschinen laufen, bei denen die Speicherbelegung des Bildschirmspeichers gleich ist. Bei Änderung des Programmes auf Printbefehle müßte man praktisch das gesamte Programm neu schreiben. Was als Möglichkeit bleibt, ist die Anpassung des Programms an das eigene System.

Ins Nirwana gepoket

Ist man zum Beispiel Besitzer eines VC 64, will ein Listing für einen VC 20 in sein Gerät eintippen und stößt dabei auf einen Pokebefehl, der als Adressierung eine Zahl zwischen 7680 und 8185 hat, so ändert man diese Zahl in eine Zahl zwischen 1024 und 2023 um, da sich innerhalb dieser Speicheradressen der Bildschrimspeicher des VC 64 befindet. Die Zahl hinter dem Komma des Pokebefehls kann gleich bleiben. Für den Fall, daß wir Bilschirmadressen des nicht erweiterten VC 20 auf den VC 64 kompilieren wollen, würde statt POKE X, Y beim VC 20 beim VC 64 POKE X-6656, Y stehen, Zu beachten ist bei dieser Übersetzung, daß der VC 64 eine andere Bildschirmaufteilung hat als der VC 20. Daher müssen die Bildschirmadressen noch sinnvoll aneinander angeglichen werden.

Selbst bei Programmen, die ein VC 20-Besitzer vor dem Erwerb einer 8- oder 16 K Speichererweiterung geschrieben hat, tritt bei direktem Zugriff auf den Bildschirmspeicher dieses Problem auf, denn mit Zuschaltung einer Speichererweiterung verändert sich auch die Lage des Bildschirmspeichers. Da die Bildschirmspeicher bei jedem Commodorecomputer in einem anderen Speicherbereich liegen, gilt das oben Gesagte für die Commodorerechner allgemein.

Das gleiche wie für den Bildschirmspeicher gilt im Grunde auch für die Farbregister. Findet man also in einem Programm, das man für den eigenen Rechner lauffähig machen will, einen Pokebefehl, der eine Adressierung aus dem Farbregisterbereich enthält, muß man diese Adresse nur gegen eine aus dem eigenen Farbregisterbereich austauschen, und das Programm läuft.

Übersichtstabelle zur Kompatibilität von Programmen:

PET 200/CBM 3000/CBM4000:

Bildschirmspeicher	: 32768 bis 33767
Farbregister	: nicht vorhanden
BASIC Start	: 1024 (HEX 0400)

CBM 8000:

Bildschirmspeicher	: 32768 bis 34767
Frabregister	: nicht vorhanden
BASIC Start	: 1024 (HEX 0400)

Nicht erweiterter VC 20:

Bildschirmspeicher	: 7680 bis 8185
Farbregister	: 38400 bis 38905
BASIC Start	: 4096 (HEX 1000)

VC 20 mit 3 K Erweiterung:

Bilschirmspeicher	: 7680 bis 8185
Farbregister	: 38400 bis 38905
BASIC Start	: 1024 (HEX 0400)

VC 20 mit 8 K oder 16 K Erweiterung:

Bildschirmspeicher	: 4096 bis 4607
Farbregister	: 37888 bis 38399
BASIC Start	: 4608 (HEX 1200)

VC 64:

Bildschirmspeicher	: 1024 bis 2023
Farbregister	: 55296 bis 56295
BASIC Start	: 2048 (HEX 0800)

Commodores im Einklang

Zum übersichtlichen Verändern von Adressen, die den Bildschirmspeicher und die Farbregister betreffen, bedient ihr euch am besten unserer Tabelle.

Die Adressen des BASIC-Starts sind zum Abtippen von Listings völlig unerheblich, da der Computer bei der Eingabe über die Tastatur automatisch den richtigen BASIC-Anfang wählt. Wichtig werden diese Adressen erst, wenn man ein Programm mit einem anderen Computersystem als dem, mit dem man es abgespeichert hat, einladen lassen will. In diesem Falle muß das Programm, das sich in den Speicherstellen aufwärts der ursprünglichen BASIC-Anfangs-Speicherstelle zu der neuen Anfangsadresse herunterkopiert werden. Dies macht man in der Regel mit einer FOR-NEXT Schleife mit Direktmodus.

Eine genauere Besprechung dieses Problems würde hier zu weit führen, folgt aber sicher in einem der nächsten Hefte von COMPUTER Praxis. ■

G.K.

BASIC FÜR BEQUEME

Befehle, die zum festen Repertoire des Users gehören, lassen sich blitzschnell in den CBM 64 bringen.

Mit Hilfe dieser Maschinenroutine lassen sich BASIC-Befehlswörter durch einfaches Drücken einer geshifteten Taste erzeugen. Drückt man z.B. SHIFT A, so erscheint auf dem Bildschirm ASC; bei SHIFT B erhält man STEP. Die weiteren Belegungen lassen sich aus der Tabelle am Ende dieses Artikels ersehen. Beim Eingeben langer Programme spart man mit dieser Routine viel Zeit.

Wer das Programm nutzen will, sollte es sofort nach dem Eintippen abspeichern, da es sich sonst von selbst löscht. Falls man sich beim Abtippen der DATA-Zeilen verschrieben hat, gibt das Programm diesen Fehler aus: es enthält eine Datensummierung!

Nach dem Start mit RUN und der — hoffentlich — erscheinenden READY-Meldung ist die Routine betriebsbereit. Wenn man also nun eine geshiftete Taste drückt, erscheint das dazugehörige BASIC-Befehlswort.

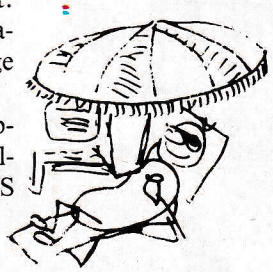
Die Routine wird mit SYS 52557 abgeschaltet. Will man sie wieder einschalten, tut man das ebenfalls mit SYS 52557. Gegen Überschreiben ist die Routine auch geschützt, und zwar dadurch, daß sie in dem vom BASIC geschützten RAM-Bereich ab hex. C000 (= dez. 49152) abgespeichert ist.

Drückt man eine geshiftete Taste innerhalb von Anführungszeichen, so erhält man das normale Graphikzeichen oder den Großbuchstaben anstelle des BASIC-Wortes, da das Maschinenprogramm die Adresse hex. D4 (= dez. 212) abprüft, die für den sogenannten QUOTE-MODE zuständig ist.

Jede Sechzigstel-Sekunde hält der C 64 an, um die Tastatur abzufragen und andere interne Aufgaben auszuführen. Diese kurzen Pausen nennt man "Interrupts" und das rechnerinterne Maschinenprogramm, das in dieser Zeit abläuft, "Interrupt Service Routine" (ISR).

Empfängt der Microprozessor das sogenannte "Interrupt-Requestsignal" (IRQ), überprüft er einige Speicheradressen, um die Startadresse (= IRQ VECTOR) zu finden, damit die ISR ausgeführt werden kann. Beim C 64 ist der IRQ-Vektor in den Speicheradressen dez. 788 und 789 enthalten, die normalerweise auf die Standard-ISR im ROM auf dez. 59953 zeigen. Da jedoch der IRQ-Vektor im RAM liegt, kann man die Adresse einer eigenen Maschinenroutine einfügen und an die normale ISR anhängen. Genau das geschieht in unserer 64-Tastaturhilfe.

Und hier nun die Tabelle der einzelnen Buchstabenbelegungen:



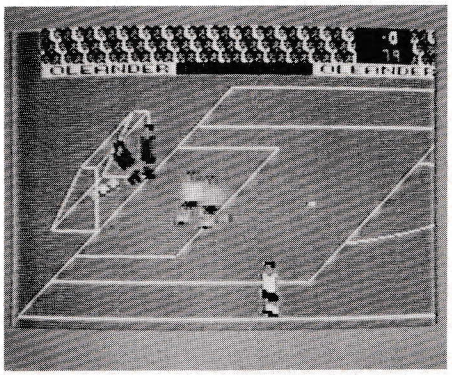
```
1000 poke 53280,0:poke 53281,0:print "{CLR}{YEL}
     {5 DOWN}"tab(10);"{RVS ON} 64 - Tastaturhilfe ":
     poke 53272,23
1010 for i=52557 to 52739:read z:s=s+z:poke i,z:next i
1020 if s<>24016 then print "{RVS ON}{5 SPACE}fehler
     in datazeilen ":goto 1040
1030 sys 52557:new
1040 end
1050 data 120,173,20,3,72,173,21,3,72,173,116,205,208,
     2,169,118,141,20,3,173
1060 data 117,205,208,2,169,205,141,21,3,104,141,117,
     205,104,141,116,205,88,96,0
1070 data 0,72,138,72,152,72,165,215,72,165,212,240,4,
     104,76,221,205,104,201,193
1080 data 144,82,201,219,176,78,56,233,193,170,189,229,
     205,162,0,134,198,170,160
1090 data 158,132,34,160,160,132,35,160,0,10,240,16,
     202,16,12,230,34,208,2,230,35
1100 data 177,34,16,246,48,241,200,177,34,48,17,8,142,
     255,205,230,198,166,198,157
1110 data 119,2,174,255,205,40,208,234,230,198,166,198,
     41,127,157,119,2,169,20
1120 data 141,119,2,230,198,104,168,104,170,104,76,49,
     234,198,169,199,134,128,129
1130 data 161,144,133,137,141,200,202,130,159,151,194,
     201,196,163,183,197,131
1140 data 135,140,158,127,0,0,0,255,255,255
```

Taste / Keyword					
a	asc	j	goto	s	str$
b	step	k	gosub	t	tab
c	chr$	l	left$	u	usr
d	dim	m	mid$	v	val
e	end	n	next	w	data
f	for	o	open	x	read
g	get	p	poke	y	restore
h	stop	q	peek	z	sys
i	input	r	right$		

Michael Schaffer

ATARI — VC 64 — EPSON — STAR
Überraschungs-Preisliste anfordern!
TEL.: 02623-6676
COMPY SHOP
DER ATARI-SPEZIALIST
5433 SIERSHAHN
BERGSTR. 18

SOFTWARE IM TEST



Soccer 2

Wie aus dem Namen ersichtlich, handelt es sich bei diesem Spiel um die Simulation eines Fußballmatches. Zu den Spielregeln muß deshalb an dieser Stelle nicht viel erklärt werden. Es handelt sich jedoch um ein 3-D Spiel mit so mancher Raffinesse.

Sobald das game geladen ist, erscheinen, nachdem das Titelbild gezeigt wurde, zwei überdimensionale große Spieler. An diesen Spielern kann man nun die Trikotfarben ausprobieren, bis man die richtige Farbkombination für den Gegner und sich selber gefunden hat. Ist die Farbwahl abgeschlossen, kann man, ebenfalls an den beiden Spielern, wählen ob man gegen einen menschlichen Gegner oder den Computer spielen will. Wählt man letzeres, wird der rechte Spieler auf dem Bildschirm durch Zahlen ersetzt, die die Spielstärke der Computermannschaft angeben.

Wenn man nun alle die eben genannten Entscheidungen getroffen hat, kann man mit dem eigentlichen Spiel beginnen. Es erscheint, wie könnte es anders sein, ein dreidimensionaler Ausschnitt eines Fußballfeldes, das den Anstoßkreis zeigt, aber keine Mannschaften. Nach einem Augenblick erscheinen aber auch die Kicker. In sauberer Aufstellung laufen die beiden Mannschaften unter dem Beifall der Massen in das Stadion ein, nehmen Aufstellung und lassen das Spiel beginnen. Wie selbstverständlich scrollt der Bildschirmausschnitt, den man betrachtet, perfekt mit dem Spielverlauf synchron. Der Spielverlauf ist hervorragend dreidimensional dargestellt, und sogar der Ball, bei dem bei den meisten 3-D-Spielen die dritte Dimension ein jähes Ende findet, ist, von der ballistischen Flugbahn bis zum Schattenwurf, glaubhaft gelungen.

Eine weitere Stärke dieses Programms, die auch in den Bereich Graphik fällt, ist die Darstellungsweise beider Torhüter. Die graphische Umsetzung ihrer Hechtsprünge verleitet einen beinahe dazu, den Gegner vor dem eigenen Strafraum anzuspielen, nur damit man mit dem Torwart eine Rettungstat nach der anderen vollbringen kann. Es ist fast ein Selbstgänger, daß die Graphik farbig und der Sound erstaunlich originalgetreu dargestellt sind.

Zusammenfassend ist Soccer 2 ein Spiel, dessen Idee nicht mehr ganz neu ist, das aber durch seine Grafik und durch seine Sound in der Lage ist, einen oder zwei Spieler für längere Zeit vor dem Bildschirm zu fesseln. Ist man also glücklicher Besitzer eines solchen Spieles, kann man eigentlich nur der Siegermannschaft zustimmen, wenn diese am Ende jubelnd mit einem Pokal unterm Arm das Spielfeld verläßt.

Graphik:

Sound:

Spielidee:

Testergebnis:

Castle Wolfenstein

Castle Wolfenstein ist der Name eines sehr ungewöhnlichen Spiels — ungewöhnlich in der Idee und der Art der Ausführung. Eigentlich ist Castle Wolfenstein in die Reihe der Adventures einzuordnen. Es bietet die Möglichkeit, ein angefangenes Spiel abzuspeichern und zu einem beliebigen Zeitpunkt fortzusetzen. Es ist aber genauso ein Actionspiel, bei dem es gelegentlich zu wilden Schießereien kommt. Die Graphik ist nicht überwältigend, aber akzeptabel.

Zeitpunkt der Handlung: das III. Reich. Ein Mann wurde von den Nazis verschleppt und zu einem Schloß gebracht. Dort gelingt es ihm, aus seinem Kerker zu entkommen, und — was für ein Zufall! — ihm fällt ein Gewehr mit zehn Schuß Munition in die Hand. Er versucht zu entwischen, indem er durch das Schloß schleicht. Doch selbstverständlich ist im Castle jeder Raum bewacht, teils durch einfache Wächter, teils aber auch durch SS-Leute, die den Weg nicht freigeben — es sei denn, man schießt sie über den Haufen.

Ferner stehen, wie sich das für ein richtiges Schloß só gehört, überall Tru-

hen herum. Darin befinden sich Schnaps, Sauerkraut, Munition, Granaten und die Kriegspläne für die Operation Rheingold. Diese Materialien helfen einem zu überleben und den Weg durch das Schloß zu finden. Die Pläne sind 'kriegswichtig' und müssen aus diesem Grunde den Deutschen entwendet und in Sicherheit gebracht werden.

Das Spiel besitzt einen eingebauten Pseudo-Sprachsynthesizer, der nicht immer ganz deutlich zu verstehen ist. Die von den SS-Leuten gebrüllten Kommandos erinnern bestenfalls an amerikanische Nazi-Filme.

Entscheidenster Nachteil von Castle Wolfenstein ist aber, daß es ein typisches Killergame ist. Wer nicht mordet, wird von den Nazis umgebracht. Das macht es zu einem äußerst bedenklichen Spiel, denn der geschichtliche Hintergrund (III. Reich) wurde wohl nur ge-

wählt, um am Geschäft mit dieser Vergangenheit teilzuhaben. Lernen kann man aus ihm nichts.

Für den CBM-64 ist das Spiel ohne Hardwareerweiterungen (Sprachsynthy etc.) lauffähig. Nützlich wären Joysticks, obwohl man Castle Wolfenstein theoretisch auch mit den Tasten spielen kann. Dann müßte man allerdings 22 Tasten bedienen! Selbst zu zweit bleiben noch 11 Tasten für jeden.

Nachteilig ist auch, daß Castle Wolfenstein den Speicher offenbar so sehr ausnutzt, daß jedesmal, wenn man von einem Raum in einen anderen geht, der Raum eingeladen werden muß. Und das dauert schrecklich lange.

Das Spiel ist wegen der relativen Files nur auf Diskette lauffähig und kostet ca. 130,— Mark.

Graphik:

Sound:

Spielidee:

Testergebnis:

Wacky Waiters

Wacky Waiters heißt ein Spiel für den VC-20 in Grundversion, das sich durch eine recht ungewöhnliche Spielidee auszeichnet: Ein Kellner muß einen Gast bedienen. So etwas ist zwar nicht gerade spannend, macht aber trotzdem Spaß, und das ist ja auch die Hauptsache. Das Spielfeld ist in fünf Etagen eingeteilt: In den oberen vier Stockwerken liegen links die Essensausgaben. Rechts sind vier Etagen leer, in der fünften sitzt der Gast, der bedient werden will. Dazwischen lauert aber die Schwierigkeit in Form von fünf Aufzügen.

Das Spiel selbst ist sehr einfach. Man muß den Gast bedienen, das heißt: ihm die Karte vorlegen, die Bestellung aufnehmen, das Essen von der Essensausgabe abholen und es dem Gast servieren. Dann auf dem schnellsten Weg zurück zur Rezeption.

Die einzige Schwierigkeit besteht wirklich nur in den besagten fünf Aufzügen. Sie zu überwinden kostet den Kellner nicht nur Zeit, sondern — wenn er nicht vorsichtig ist — den Job.

Es gibt aber auch kleine Tricks, mit denen man das Trinkgeld erhöhen kann: Man kann zum Beispiel, sobald der Gast gerufen hat, im letzten Aufzug auf den nächsten Kunden warten. Dadurch spart der Spieler sich den Weg zum Kunden hin. Ganz ohne Tücken geht es dann aber auch nicht, denn ein zweiter Kellner geht quer durch das Spielfeld hindurch. Man wird entlassen, wenn sich beider Wege kreuzen.

Wacky Waiters hat uns vor allem aus zwei Gründen recht gut gefallen. Zum einen, weil es kein Action- oder Ballerspiel ist, zum anderen aber weil es besonders einfach zu handhaben ist. Zur Bedienung braucht man nur zwei Tasten, und die können noch durch einen Joystick ersetzt werden.

Trotz seiner Einfachheit ist Wacky Waiters kein Spiel, an dem jeder, der älter als 6 Jahre ist, die Lust verliert.

Das Spiel ist auf Cassette im Handel zu haben, Ladenpreis ca. 30,— Mark.

Graphik:

Sound:

Spielidee:

Testergebnis:

Programme ...

Für die Commodore-Rechner wurden hier dreizehn interessante BASIC-Programme erstellt und mit Problembeschreibung, Lösung und Programmerläuterung so aufbereitet, daß Anfänger und Fortgeschrittene gleichermaßen davon profitieren können.

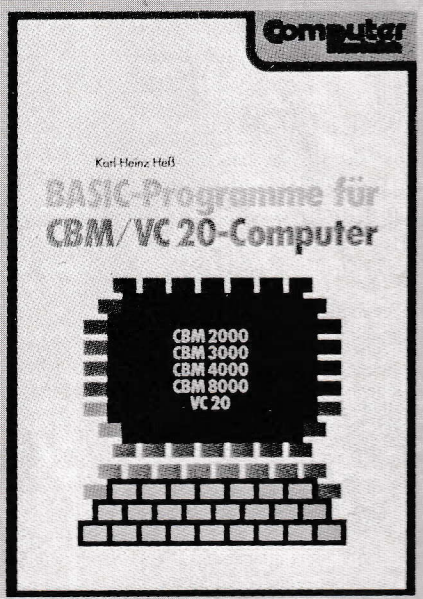
Etiketten-Beschriftung und Nadelansteuerung des CBM-Printers 3022, Lottozahlenermittlung und das Ausfüllen von Banküberweisungen, Kopfrechentraining, die Darstellung horizontaler Balkendiagramme und noch einiges mehr sind die Themen.

Die Listings wurden verständlich beschrieben und gut dokumentiert. Durch die kleine und teilweise ausgefranste Schrift sind sie leider nicht immer ganz so gut lesbar.

Sozusagen als Nachschlag gibt's noch sieben kleine Fünf-Minuten-Programme "ohne Worte", die in erster Linie Spaß vermitteln sollen.

Die Programme dieses Buches laufen auf allen Commodore-Rechnern.

Thok



BASIC-Programme für CBM und VC 20 Computer

Hess
Verlag Markt und Technik
32,— DM
150 Seiten
ISBN 3-922120-28-8

Micro-Markt

Eine private Kleinanzeige (max 5 Zeilen von je 30 Druckzeichen einschließlich Satzzeichen und Wortzwischenräumen) kostet DM 2,— pro Durckzeile. Also höchstens DM 10,— inkl. Mehrwertsteuer.

Eine gewerbliche Kleinanzeige kostet DM 10,— pro Druckzeile, in diesem Fall zuzüglich Mehrwertsteuer. Eine Aufnahme von Chiffre-Anzeigen ist leider nicht möglich!

Den Anzeigenpreis entrichtet ihr am besten per Verrechnungsscheck direkt mit eurem Inserat an uns.

Wenn ihr möchtet, könnt ihr aber auch das entsprechende Bargeld dem Auftrag beifügen oder das Geld auf unser Konto bei der Deutschen Bank, BLZ 200 700 00; Konto-Nr. 410 63 81 überweisen.

Um eure Anzeige in der nächstmöglichen Ausgabe von COMPUTER Praxis zu veröffentlichen, braucht ihr nur den auf dieser Seite abgebildeten Coupon auszufüllen und auszuschneiden. Dann in einen ausreichend frankierten Umschlag stecken und an uns abschicken.

Unsere Anschrift lautet:
COMPUTER Praxis
Marshall Cavendish Ltd. Int.
Loretta Arnold
Heidenkampsweg 76 A
2000 Hamburg 1

Atari

Verkaufe Atari 600 XL + Programmrecorder, 3 Mon. alt, B. mit Gar, Bestzustand, DM 620,— HA 02331 / 575 93

Erweitere Atari 400 auf 48 K für 160 DM. 1/2 Jahr Garantie. Rette defekte »52 K-Geräte«. Tausch von über 800 Progr. Disk. Bei S. Schmeling. 0431 / 54 25 43

Verkaufe: Spielmodule und Cassetten für den Atari 400/800 Module: Donke K., Joust, Pol Pos. usw. Cass: 40 St. Tel. 06894 / 528 07

50 DM Belohnung für Listing Suche Programm f. Bundesliga-Tabelle auf Atari 600. Das beste wird bezahlt. M. Kluckow Angelastr. 19, 4830 Gütersloh

Suche Atari-Games die ich veröffentlichen kann. Zahle gute Provision. Game-Cassetten an: Andreas Hettinger, Taunusstr. 42, 6100 Darmstadt.

Atari Software-Gazette Exclusive Spielbeschreibungen mit Skizzen und Photos — heiße Tips und rasante Ideen. 6,— DM an Diskeater Postf 602465 2 HH 60

Commodore

CBM64/VC20 — Listschutz Info bei Walter Trawnitschek 1020 Wien, Böcklinstr. 24/2/32

C-64 Haushaltsprogramm, sehr komfortabel, 11 Ktn., 900 Bchg PM., gute Grafik. Kass. DM 20,— Info 80 PF. F.W. Hornung Seewiesen 1, 7410 Reutlingen 28

Neueste Software **C-64** aus USA. Katalog 3,— in Briefm. Bei: **American Soft,** Postlager-Karte 98815A, 2300 Kiel 1 **02400PRG**

SUPERSPIELE und Programme Tausch od. gegen Unkostenbeitrag. Info kostenlos! Es lohnt sich! D. Waltemathe, H. Heinestr. 4 3260 Rinteln 8, Tel. 05751 / 56 07

Suche Bauwesen, Baustatik, — Verwaltung und Organisation für C 64. Kiewitt, Klapperstraße 100, 4300 Essen 14

CBM64 Textverarbeitung Texti2+ Randausgleich, dt. Sonderzeichen für Disk + 1525 Drucker 50 DM-Schein an M. Ferrari, Sandbuschweg 22, 3500 Kassel

VC 20 Programme für alle!!!! Liste kostenlos! 05722 / 246 42

C64 Supersoftw. nur neue Spiele. Kein Schrott! Nur Topspiele IG-Soft, Homburger Str. 35, 6365 Rosbach. Liste:1,30 DM Brm.

Super Angebot — C 64 + TI99/4A — Top-Spiele 83/84 für wenig Geld. Bei J. Ufermann, Nußbaumweg 9, 4620 Castrop-R. Freiums. 80 Pfg.

VC20/C 64 Recorderinterfacebaus. mit Geh. und gen. Anleitg. 25 DM G. Schons, Malteserweg, 5305 Alfter

Achtung!!! 64-er Spielefreaks: Verkaufe meine Spielesammlung (ca. 20 St.) auf Diskette. Nur das Feinste vom Feinen (Miner, Defender) 300M, J. Müller, 08192 / 14 17

VC 20 SPITZENPROGRAMME Superpreise, Info gratis: Santoro, U-Mühlewiesen, 7896 Degernau

Suche f. VC-20 Floppy u. Druck. Suche Prog. f. VC-20 (Phoenix) u.a. Manuel Wendt, Rissener Landstr. 6, 2000 Hamburg 55

VC-20 + 32 K RAM + 90 Programme + 45 Modulprgrme. (Choplifter Shamus, EXBASIC, Forth) NP über 1500 DM für nur DM 390!!! Herrnberger, Tel: 08431 / 88 64

C-64 Programme gegen Unkosten oder Tausch abzugeben. H-J Schmitz, Roesrather Str. 457 5000 Köln 91, Tel: 0221 / 86 24 57

Super CBM 64 Prg's total billig. Immer nur das Neueste nur Disk. Postlagerkarte 098858A 23 Kiel 1 Programmliste kostenlos!!! Ich bin auch zum Tausch bereit

Tausche Software für CBM 64 Diskette und Kassette T. Schumm, Rob-Schumann-Str. 12 7180 Crailsheim, Tel. bevorzugt Tel.: 07951 / 221 90

Sinclair

Sinclair Spectrum 48 mit Dk' Tronics-Tast. und 2 Bücher für 650,— DM zu verkaufen. 4 Mon. alt. Koehler Tel. 040 / 644 01 86 ab 20 Uhr

Spectrum 48 K-Datafile 1 Dateienprogramm: Suchen, listen, sort, dt. Zeichen, ausf. Handbuch. DM 50,— Scheck/NN-Kniese, Alt-Rödelheim 40, 6000 Frankfurt 90

ZX81-Software, Große Auswahl an Prgmen, Tips, Infos, Spielen, Amateurfunk-Softw. Liste gegen Rückporto. Michael Schramm, Freiligrathstr. 5, 2300 Kiel 1

Copy kopiert jedes Spectrum-Prgm. Copy auf Cassette + Catalog-Prgm + Listing/Beschreibung nur DM 16,— Michael Schramm, Freiligrathstr. 5, 2300 Kiel 1

Texas Instruments

Suche Extended-Basic-Modul und Joysticks Tel. 07240 / 53 00

Suche Extended BASIC-Modul für TI99/4A. Kaufe event. auch andere Programmiersprachen. Andreas Göring, Middelkerker Str. 74, 7505 Eftlingen

***TI-99/4A *HUGGYSOFT * TI-99/4A* ***
*Spitzensoftware für Ihren TI *
*Info gegen Freiumschlag an *
HGS-Soft Burgstr. 14 498 Bünde*
Demnächst auch SV-328 Software

Suche TI 99/4A + Ext-BASIC und RS 232 (V24)-Extern, nur TI-original, rufe zurück nach 20h Tel. 0241 / 710 67 (von 8.30—16.00)

TI Peripherie-Box ohne Erweiterungs Karten zu verkaufen. Manfred Lichtenberg Peterstr. 26, 425 Bottrop, 02041 / 226 61 ab 18 Uhr

TI99/4A TI99/4A TI99/4A TI99/4
Suche **Dringendst *Minimemory*** Zahle Höchstpreis!!!
Dirk Bernhard, Schillerstr. 18 6699 Freisen 1, Tel. 05855 / 63 62

TI99/4A 1A Software Bitte an: R.M. Frazer, Kessenicher Str. 13, 53 Bonn. Demo-Progr. geg. Rückporto

Biete noch wenige aber günstige Hard- + Software an. Alle 50 neuen USA Module. TI-Forth, Pilot + CP/Metc. Für 3,— 20 S. Katalog + 1 Listing, Newsletter. A. Berti, Zielstr. 9, CH-8400 Winterthur

SUCHE EXTENDED BASIC! 089 / 793 22 73

Suche Extended BASIC, Schach, 32 K RAM. Tel. 0211 / 61 34 52

Tausche/verk. TI99-Prg., auch Ext.-BASIC. Tel.: 02161 / 64 24 26

****
Software superbillig
Internationale Spitzenprogr. ab DM 4,— **Liste bestellen* G. Mahlig, Auf dem Heidenberg 19, 5000 Köln 60, Tel. 0221/712 23 58

TI99/4A, Extended BASIC, Record. + Kabel, mehrere Module (Parsec, TI-Invader, .., Datenverw., Analyse) Anwender- u. Spiele-Softw., Bücher + Literatur Tel. 06131 / 5 86 94

TI99: Spannende Aktions- u. Strategiespiele, kein Massenramsch, sond. wenige ausgefeilte Pgmme. Ausf. Info g. Rückp. Dieter Taube, Dingelstedtwall 16, 326 Rinteln

TI99/4A Super Ext. Spiele z.B. Dchungle Pit, Star Raider, Atlantis, Schlümpfis Abenteuer, Dracula, Alpha Alarm, Utopia, Supersound, Graphik. Tel. 02226 / 74 40

Suche Hard- und Software. Speziell Ext. BASICmodul. TH. Thoene Altenhofstr. 11, 8070 Ingolstadt

Suche Extended-Modul!
Michael Wirthmann, Brunnenweg 8 8786 Rieneck

Verschiedenes

SHARP PC-1211 + Interface CE-121 zu verkaufen
Tel. 02195 / 23 92

Suche nebenber. Tätigk. für TI99/4A + EX-BAS + Drucker + 32K + Floppy + TI-Writer + Mini-Mem + Dat. Verw. + Statist. usw. ✻W.F. Grasmaier✻, Max-Schott-Str. 16, 8070 Ingolstadt

Auto-Cassettenradio
Hitachi Digital 6, 1/4 J. alt, sowie 2 Booster-Equalizer (7-fach, 2x30W), jeweils 50,—. Rainer Krause, Tel. 040 / 54 59 25. Öfter versuchen.

Suche freiberufliche Tätigkeit an Entwicklung von Hardware, Meßdatenerfassung etc. 05103 / 132 29 nach 18.00 Uhr

Stereoanlage: Plattensp. Dual CS 604 m.M75, Verst. Hitachi HA-5300, Cass. Technics M65, Equalizer Technics, Preis VHS. Michael Krieger, Tel. 04193 / 71 66, 19.00 bis 22.00 Uhr.

Suche Software CPM usw. für Indecdata Superbrain. 05031 / 132 29 abends.

Kontakte

C-64 User Club Germany!
Was wir bieten? Clubmagazin, Softwarebank, Servicetelefon und mehr... Info anfordern (3,— in Briefm.) beim C-64 User Club Germany, Hildesheimer Str. 388, 3 Hannover 81

SPECTRUM-User-Club nimmt neue Mitglieder auf. Monatl. Info, Tips Erfahrungsaustausch, Probleme weitere Infos von: Spectrumpool Postfach, 3130 Lüchow

Hallo Atari User! wer einen Atari-Computer hat und Lust hat, einen Club zu gründen, meldet sich bei Axel Freudenstein
Tel: 05601 / 23 22 ab 18 Uhr.

Gewerbliche

Preisbrecher
Ab 410,— Laufw. von BASF, Shugart, DISK-Stationen, Comput. von Apple, Tandy, auch Drucker, Festplatten, u.a.m., Cp.-Discount
B. Vogel, PF 2609,
8960 Kempten

Computertechnik + Programmierung
Ausbildung durch bewährte und anerkannte Fernlehrgänge, Technik, Hard/Software, Assembler, Maschinensprache, BASIC. Ein Übungs-Computersystem wird mitgeliefert. Information: Fernschule Bremen 2800 Bremen 34/
PF 7026/8-163

Listing-Korrekturen zu Heft 5/84

Leider ist es uns doch nicht gelungen, unser Versprechen einzuhalten, daß unsere Listings fehlerfrei seien. Das lag zum einen an einem unentdeckten Problem in unserem Formatierungsprogramm, zum anderen haben wir beim Korrekturlesen einige Dinge schlicht übersehen. Wir bitten euch um Entschuldigung und liefern hier die Berichtigungen:

17 + 4, S. 14: im Listing fehlen folgende Zeilen:

```
1080 POSITION 31,2:?
     "WERT:"; KWS;" "
1090 SPS=SPS+6:RETURN
1100 REM COMPUTER NIMMT
     KARTEN
1110 POSITION 1,15:? KC1$
1120 GOSUB 840:GOSUB 900
1130 IF KWC> =17 THEN
     RETURN
1140 GOTO 1120
1150 REM WAHLTAFEL
1160 FOR V=1 TO 50:NEXT
     V: ZEI=5
1170 POSITION 31,5:?
     "KARTE␣␣␣"
```

Player/Missile Editor, S. 19:
Dieses Programm läuft NICHT mit 16 K!

Breakout, S. 36:
am Ende der Zeile 390 muß es heißen: goto 290.

Trafalgar, S. 52:
hier hat unser Formatierungsprogramm in einigen Zeilen das letzte Zeichen verschluckt. Fügt bitte ans Ende der jeweiligen Zeilen die unten aufgeführten Zeichen an.

Zeile	anfügen	Zeile	anfügen
76	1	435	5
175	1	645	1
232	0	707	Y
284	)	762	)
352	)	2055	1

Schweinchensuche, S. 66:
in den Zeilen 3470, 3690, 4090 und 4210 muß es statt CALL HCHAR heißen: CALL GCHAR.
Das ¢-Zeichen in den Zeilen 4840 und 4850 muß durch ein ^ Zeichen ersetzt werden.

Auftragscoupon COMPUTER Praxis

Veröffentlicht bitte im nächstmöglichen Micro-Markt von COMPUTER Praxis folgenden Anzeigentext

unter der Rubrik

☐ Hersteller: _____ ☐ Kontakte ☐ Verschiedenes

Meine Anzeige ist:
☐ eine private Anzeige (DM 2,— incl. MwSt. je Zeile)
☐ eine gewerbliche Anzeige (DM 10,— zzgl. MwSt. je Zeile)
☐ Verrechnungsscheck liegt bei ☐ Bargeld ist beigefügt
☐ Betrag ist auf die Kto.-Nr. 410 63 81, Deutsche Bank, BLZ 200 700 00 zu überweisen
Mit der Unterschrift bestätige ich, daß die angebotene Software keine Raubkopien sind.

Datum Unterschrift

LESERBRIEFE

Ausgewogen

Ich bin Besitzer eines Computer/Fiat 500, sprich ZX81, und hatte recht wenig von Euren garantiert fehlerfreien Listings, denn anscheinend wurde der ZX81 nicht für würdig gehalten, mit einem Listing bedacht zu werden. Wirklich, Ihr solltet die Zahl der ZX81'ler nicht unterschätzen. Wenn schon Listings für Sinclair, dann müßte der Platz für Spectrum und ZX81 auch 50:50 aufgeteilt werden.

Was mir noch aufgefallen ist: die Verteilung der Seitenzahlen auf die einzelnen Systeme ist nicht ganz ausgewogen. Sicherlich wird man nicht jedem System die gleiche Seitenzahl zukommen lassen können — nur leidlich ausgewogen sollten die Proportionen sein.
Karlheinz Müller, Kriftel

Hallo Karlheinz!

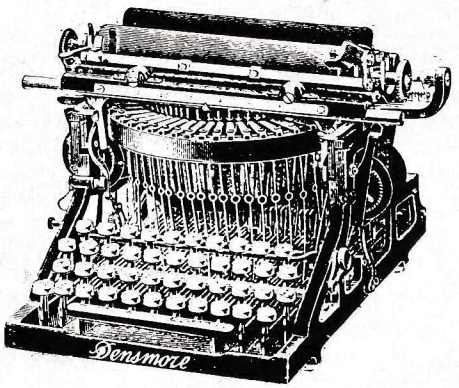
Daß Du mit dem Mengenverhältnis in Heft 4/84 nicht ganz einverstanden bist, kann man verstehen. Uns liegen leider allzu wenige Programme für den ZX81 vor, und die wenigen, die wir haben, wollen wir nicht ohne Test vorstellen. Wir hoffen aber weiterhin auch auf Programme unserer Leser! Wir streben nicht an, unbedingt ausgewogen zu sein. Wir finden es sinnvoller, mal über das eine, mal über das andere System ausführlicher zu berichten, je nachdem, wo es besondere Neuheiten oder Probleme gibt. Das dürfte sich dann von Heft zu Heft ausgleichen.
Die Redaktion

Leserbeiträge sind uns willkommen

Ich finde Ihre Zeitschrift ausgesprochen gelungen.

Ich würde mich gern am Spielbeitrag beteiligen, sagen Sie mir bitte, wie ich das anfangen soll?

Dann hätte ich noch einige Verbesserungsvorschläge, z.B. Fotos nicht so groß, Tips von Anwendern für Anwender, Marktneuheiten usw.
Andreas Schwarz, Leverkusen



Lieber Andreas!

Vielen Dank für das Lob.

Mit den Leserbeiträgen funktioniert das bei uns so: Wer ein Programm geschrieben hat, soll es uns schicken, möglichst mit einer kurzen Beschreibung. Dazu brauchen wir eine schriftliche Erklärung, daß der Einsender auch wirklich Urheber des Listings ist. In unserer Redaktion wird das Programm dann von dem Experten für das jeweilige System getestet. Wenn es uns geeignet erscheint, melden wir uns sofort. Es versteht sich von selbst, daß wir Eure Programme nicht aus der Hand geben.

Mit den Verbesserungswünschen ist das so eine Sache, oft kommen Sonderwünsche einzelner Leser. Wenn aber gleiche Vorschläge öfter gemacht werden, besprechen wir diese und werden uns gern darauf einstellen.
Die Redaktion

Modem für TI

Ich habe gelesen, daß man an den TI 99/4A ein Modem anschließen kann. Wie kann ich das Modem ohne Modulbox von TI an meinen Computer anschließen? Was brauche ich dazu? Wo kann ich gegebenenfalls das benötigte Gerät kaufen, was kostet es?
Dirk Gerspacher, Kaiserslautern

Hallo Dirk!

Grundsätzlich kann ein Modem an jeden Computer angeschlossen werden. Die meisten werden über eine RS-232-Schnittstelle gesteuert. Die RS-232 ist zu haben bei:
atronic, Bunke & Vernunft oHG, Postfach 70 01 24, 2000 Hamburg 70 Tel.: 040 / 668 22 56
oder
Klein Computer, Münzstraße 34 a, 5600 Wuppertal 2, Tel.: 0202 / 50 14 05

Der Preis für die Schnittstelle liegt, je nach Ausführung, bei 300,— bis 450,— Mark. Ein TI-kompatiblles Modem soll demnächst von atronic auf den Markt gebracht werden.
Die Redaktion

Bitte bessere Spiele

Ich spreche Euch extra so an, weil man bei Euch irgendwie merkt, daß dieses neue Computermagazin von Freaks für Freaks gemacht wird.

Eure lockere Einleitung gefiel mir gut, auch der Schreibstil bei verschiedenen anderen Beiträgen. Allerdings gibt es auch ein wenig Kritik, so z.B. die Spiel-Rezensionen. Es gibt doch wirklich andere (und bessere) Spiele als "Cyclons" und "Pacacuda". Ich besitze selber beide Spiele und muß sagen, daß sie nicht zu meinen Lieblingsspielen zählen.

Auch sollte man vielleicht das Ganze noch ein bisserl durch Zeichnungen oder so auflockern. Die allgemeinen News fehlen mir auch so ein bißchen, aber sonst? Echt stark das Heft, beim heiligen Computerus!

In diesem Sinne wünsche ich Euch auch weiterhin "gut hack".
Kurt Labude, Kassel

Servus, Artgenosse Kurt!

Haben wir so viel Lob wirklich verdient? Würde uns freuen!

Was die Spiel-Rezensionen angeht: da sind wir ein bißchen abhängig von den Herstellern, die uns die meisten Spiele zum Testen zur Verfügung stellen, wir können ja nicht alle kaufen! Und für unser erstes Heft war die Auswahl einfach noch recht klein.

Aber wir geben uns Mühe, bessere Spiele zu finden.
Die Redaktion ■

APPLE II
ATARI
CBM

Drei Standardwerke zur maßgeblichen Unterstützung für den Benutzer des jeweils angesprochenen Homecomputer-Typs. Mit dieser Begleitliteratur werden Sie Ihren Computer erfolgreich einsetzen und wirklich alle Möglichkeiten nutzen können, die in dem System verborgen sind.

Apple II Anwenderhandbuch. L. Pool. DM 56,–
MEIN ATARI COMPUTER. L. Pool, McNiff & Cook. DM 59,–
CBM Computer Handbuch. Osborne/Donahue. DM 59,–

Die angegebenen Preise sind die Ladenpreise.

te-wi

te-wi Verlag GmbH
technisch wissenschaftliche Elektronik-Literatur
Theo-Prosel-Weg 1, 8000 München 40

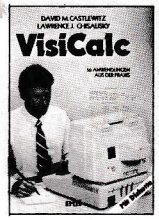
Weiterführende Literatur...

Apple II PASCAL
Unentbehrlich für alle, die die Programmiersprache PASCAL lernen wollen und Zugang zu einem Apple II-Computer haben.
A. Luehrmann/H. Peckham. DM 59,–

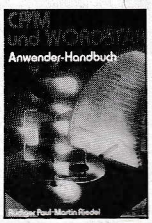
77 BASIC-Programme
77 Kurzprogramme, die finanztechnische, mathematische, statistische und verschiedene | allgemeine Aufgaben mit Programmbeispielen behandeln.
L. Pool/M. Borchers. DM 39,–

Apple Maschinensprache
Für BASIC-Programmierer der einfachste Zugang zur Muttersprache des Apple. Wesentlich schnellere Maschinenprogramme, direkte Manipulation des Mikroprozessors 6502 im Apple – als Brücke dorthin benötigt dieses Buch nur die drei BASIC-Befehle, POKE, CALL, PEEK. D. Inman/K. Inman. DM 49,–

6502 – Programmieren in Assembler
Dieses Buch behandelt ausführlich die Assemblersprachen-Programmierung für den weitverbreiteten Mikroprozessor 6502.
L. Leventhal. DM 59,–



VisiCalc – 50 Anwendungen aus der Praxis
Dieses Buch enthält eine Sammlung von 50 der häufigsten VisiCalc-Anwendungen in Wirtschaft und Privatbereich. Alle Berechnungen und die Darstellungsform sind auf der beigefügten 5 1/4"-Diskette gespeichert. Bei Bestellung Computertyp angeben.
D. Castlewitz/L. Chisausky. DM 79,–



CP/M und WordStar
Ein Standardwerk für dieses Betriebssystem, das dem ständig wachsenden Kreis von Mikrocomputer-Anwendern eine fundamentale Einarbeitungshilfe bietet. Paul/Riedel.
DM 29,80

In Vorbereitung 2. Quartal 1984: C64 Computer Handbuch, DM 56,–
IEEE488 -Buch und Steckmodul für C64, DM 239,–
IBM-PC Anwenderhandbuch, DM 59,–

Die angegebenen Preise sind die Ladenpreise.

tm 3838

Computer Clubs

Zukunftsbewältigung will der Computerclub "Floppy" betreiben — für Vergangenheitsbewältigung gibt's den Heimatverein ...

Auch Buxtehude (da, wo die Hunde mit dem Schwanz bellen) hat seinen Computerclub. Jung ist er, gegründet am 20. März 1984. Und hat schon 50 feste Mitglieder. Die Lokalpresse, die natürlich ein offenes Ohr für so zukunftsweisende Aktivitäten hat, jubelt, daß es wöchentlich mehr werden! Kein Wunder, denn, wie die Floppy-Leute sagen, ist Norddeutschland eine Computerwüste.

Starke Sprüche haben sie drauf, aber das ist nicht das Besondere. Der Floppy-Club ist der einzige uns bekannte Computerclub, der nicht nur ein eingetragener Verein ist, sondern der von den Behörden als gemeinnützig anerkannt wurde. Man weiß ja, daß man Außergewöhnliches leisten muß, um sich 'gemeinnützig' nennen zu dürfen, aber man weiß auch, daß ein gemeinnütziger Verein Spenden entgegennehmen darf...

Die außergewöhnlichen Leistungen der Floppy-Leute bestehen darin, daß sie, wie sie sagen, etwas für die 'Volksbildung' tun. Das ist sicher auch nötig, denn an den Buxtehuder Schulen gibt es insgesamt nur 4 Computer — viel zu wenig für all die Interessierten. Und auch die Volkshochschule hilft da wenig: sie hat gar keine Geräte, außerdem wären die Kurse teurer als der Vereinsbeitrag (DM 5,— für Jugendliche, DM 10,— für Erwachsene, je Monat). Man hat also eine echte Marktlücke entdeckt und nutzt sie aus — was ja auch legitim ist.

Gemeinnützigkeit: ist Ermessenssache

Ganz so einfach war das mit der Gemeinnützigkeit dann aber doch nicht. Ein Verein, der als gemeinnützig anerkannt werden will, muß zunächst eine Satzung entwerfen, diese dann an das zuständige Finanzamt schicken und hoffen ... Das Finanzamt fand die Idee aber gar nicht so weltbewegend, es verlangte einige Änderungen in der Satzung. Wie günstig war es da doch, daß im Vorstand des Vereins ein Steuerberater sitzt, der wußte, daß man vielleicht über eine Kontaktperson in Hamburg-Harburg an die Hamburger Behörden herantreten könnte, vorausgesetzt, man erklärt Hamburg zum 'Hauptsitz' des Clubs. Siehe da, es klappte, das Finanzamt Hamburg fand die Idee gemeinnützig genug.

Und wie ist das mit den Spenden? Nun, bis jetzt hat der Floppy-Club noch keine bekommen, aber man plant eine großangelegte Brief-Aktion, von der man sich einiges erhofft. Peter Gronemeier, seines Zeichens Bereichsleiter Öffentlichkeit und von Beruf Steuerberater, setzt große Hoffnung auf die Banken und größeren Firmen von Buxtehude, die mit einer solchen Spende ja auch Werbung machen könnten...

Ja, und wenn das große Geld dann erst einmal da ist, will der Club Geräte anschaffen und, wenn's reicht, vielleicht auch eine Wohnung mieten, denn noch benutzt man einen Raum in der Kanzlei Peter Gronemeiers. Am schönsten wäre es natürlich, man könnte eine leerstehende Schule mieten ... Gewaltige Plä-

Andres, 17 J., unterrichtet BASIC in einer der Schnuppergruppen.

In Gruppen mit so wenigen Mitgliedern fällt das Lernen jedem leicht.

ne, nur das Geld ist noch nicht da.

Wie macht denn der Verein das nun mit der 'Volksbildung'? Auf den ersten Blick wirkt es perfekt, durchorganisiert bis ins kleinste Detail. Nach Aussage des 1. Vorsitzenden, Thomas Müller, gibt es momentan etwa 50 feste Mitglieder, dazu kommen noch ca. 20, die noch nicht so recht wissen, ob sie dabeibleiben wollen. Und weil man so viele Leute nicht unter einen Hut bekommen kann, hat man sich in Gruppen zu den verschiedensten Themen aufgeteilt.

Da gibt es zunächst die 'Schnuppergruppen' für alle die, die noch nie mit einem Computer gearbeitet haben. In diesen Gruppen wird von der Pike auf BASIC gelernt. Geleitet werden die Schnuppergruppen von Schülern, die schon programmieren können. COMPUTER Praxis war bei einer Sitzung einer Schnuppergruppe dabei — und wir hatten den Eindruck, daß der Leiter es sehr genoß, einmal in die Lehrerrolle schlüpfen zu dürfen.

Wer mehr weiß, darf Lehrer sein

Grundsätzlich gilt für die verschiedenen Gruppen, daß mindestens 3—4 Mitglieder vorhanden sein müssen, von denen einer die Rolle des Leiters übernehmen kann.

Weitere Themen, mit denen sich die Floppy-Leute beschäftigen, sind z.B. der Bau des NDR-Computers, der Aufbau einer Programm-Bibliothek, das Erlernen anderer Programmiersprachen,

Nimmt seine Sache ernst: Andreas, 16 J. ist 2. Vorsitzender

Computervernetzung etc.

Überhaupt: an Ideen für neue Gruppen mangelt es keineswegs: man könnte sich mit Computer-Musik beschäftigen, Roboter bauen, Wettbewerbe organisieren ... Und als Achim Müller, Bereichsleiter Wissenschaft, einen alten Lochkartenleser als Geschenk angeboten bekam, hatte er sofort die Idee, eine Gruppe "Computergeschichte" ins Leben zu rufen. Ob das wohl wen interessiert?

Man sieht: das ganze System ist ungeheuer ausbaufähig angelegt, die Vereinsmitglieder rechnen mit großem Zulauf — allerdings nicht von den Mädchen des Umlandes: "Die können sich wohl nicht vorstellen, daß sowas Spaß macht! Die gehen wohl lieber in die Disco," mußten wir hören.

Aber die perfekt aussehende Organisation läßt auch andere Gedanken aufkommen, besonders, wenn man Zuständigkeitsbezeichnungen hört wie "Bereichsleiter", "Jugendwart", "Schriftführer". Sicher, von einer gewissen Vereinsgröße an kommt man um die Verteilung von Zuständigkeiten nicht herum — aber ein Beigeschmack von deutscher Vereinsmeierei drängt sich einem hier doch auf.

Wenn man dann noch an die gemütliche Runde denkt, die sich zum Erfahrungsaustausch trifft, liegt der Vergleich mit den Veteranen vom Fußballverein vielleicht nicht so fern.

Und wie das mit den meisten deutschen Vereinen so ist, nehmen auch die Mitglieder des Computer-Clubs Floppy ihre Sache ungeheuer ernst. "Wir sind Computerfans — aber keine Telespiel-Enthusiasten" erklärt der 1. Vorsitzende. Jawohl, gelernt wird, und nicht gespielt. Und abgekupfert schon gar nicht. Das wäre ja schließlich auch nicht im Sinne der gemeinnützigen Volksbildung, die übrigens auch Eltern in Anspruch nehmen (manchmal allerdings mit dem ausdrücklichen Wunsch, nicht in die gleiche Gruppe gesteckt zu werden, wie ihre Sprößlinge!).

Wie gesagt: das Konzept des Vereins wirkt professionell. Ein bißchen zu professionell, wenn man bedenkt, daß der 1. Vorsitzende 17 Jahre alt ist, der 2. Vorsitzende 16. Bei aller Demokratie — ob nicht vielleicht die Stimme der erwachsenen "Bereichsleiter" doch ein kleines bißchen mehr Gewicht haben?

Mag sein, daß wir uns mit unserer Einschätzung irren, daß der Eindruck, den wir aus einem einstündigen Gespräch gewinnen konnten, trügt. Eins steht auf jeden Fall fest: Wenn sich so viele Interessenten für ein solches Projekt finden, war es die höchste Zeit, daß es in Angriff genommen wurde. ■

W.K.

Allgemeine Grundlagen der Dateiverarbeitung

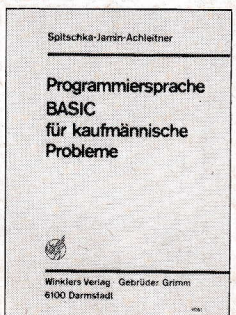
208 S., zweifarbig, Best.-Nr. 4056C, 20,20 DM
Lösungsheft, 32 S., Best.-Nr. 4057C, 4,20 DM

Aus dem Inhalt:

- Sequentielle Datei, gestreut/wahlfreie Datei, index-sequentielle Datei, gekettete Datei
- Suchen, Mischen, Sortieren, Gruppieren der Datei
- PAP, Struktogramm und algorithmischer Entwurf zu über 50 Abläufen mit Dateizugriff

Disketten für verschiedene Betriebssysteme sind zu diesem Buch lieferbar.
Nähere Informationen auf Wunsch.

Programmiersprache BASIC für kaufmännische Probleme



95 Seiten, 2.[1] Auflage, Best.-Nr. 4051 C, 11,– DM
Lösungsheft, 32 Seiten, Best.-Nr. 4052C, 4,20 DM

- Grundlagen der Programmiersprache BASIC, Einführung in verschiedene Sortierlogiken und in die Dateiverarbeitung
- Methodisch-didaktische Aufbereitung mit PAP, Listing, Erläuterungen, praxisbezogenen Beispielen
- 100 Beispiele und Aufgaben

Winklers Verlag · Gebrüder Grimm
Postfach 11 11 20 · 6100 Darmstadt

Eine solide Basis

In der zweiten Folge
unseres BASIC-Kurses für Anfänger
erfahrt ihr noch einige wissenswerte
Einzelheiten über das Programmieren.
Und dann geht es richtig los:
Wir schreiben unser erstes Programm.

Etwas erstaunen ließen wir euch vielleicht in der ersten Folge unseres BASIC-Kurses. 64 K entsprechen 65.536 einzelnen Speicherplätzen. Dort etwas wiederzufinden, müßte in etwa wie die berüchtigte Stecknadelsuche im Heuhaufen funktionieren. Aber BASIC soll doch einfach sein...

Muß der Programmierer alle 65.536 Speicherstellen kennen?

Gott sei Dank nicht! In der BASIC-Sprache benutzt man die Nummern der Speicherstellen ja gar nicht, sondern gibt ihnen Namen. Diese Namen nennt man symbolische Adressen (labels). Man sagt also z.B. nicht "Speichere die Zahl 3 in der Speicherstelle 43122", sondern man gibt den Befehl "Speichere die Zahl 3 in einer Speicherstelle mit dem Namen 'A'". Das heißt dann in BASIC: LET A=3. Meistens weiß der BASIC-Programmierer noch nicht einmal, welche wirkliche Adresse sein Speicherplatz 'A' tatsächlich hat.

Für ein größeres Programm braucht man Hunderte solcher symbolischer Adressen. Da einfache BASIC-Versionen, wie sie für Heimcomputer verwendet werden, nur eine begrenzte Zahl solcher Adressen zulassen (z.B. nur die Buchstaben A bis Z), sind dem Benutzer eines Heimcomputers schon von vornherein Grenzen gesetzt.

Womit sollte man beim Programmieren beginnen?

Wie wir schon in unserem Teil 1 besprochen haben, besteht der erste Schritt darin, daß man sich überhaupt klar wird, was man mit dem Programm erreichen will.

Nehmen wir an, wir wollen ein kleines Übungsprogramm schreiben, in dem die wichtigsten BASIC-Befehle vorkommen, so daß man einen ersten Eindruck von der BASIC-Sprache und vom Wesen der Programmierung bekommt. Dafür müssen wir uns eine entsprechende Aufgabe stellen und anschließend all die verschiedenen Lösungsmöglichkeiten durchdenken, die uns für diese Aufgabe zur Verfügung stehen.

Als erstes setzt schon unser Computer, also die uns zur Verfügung stehenden Hardwaremöglichkeiten, unserem Tatendrang Grenzen. Wir wollen deshalb erst einmal von der einfachsten Grundausrüstung, bestehend aus Zentraleinheit, Tastatur und Bildschirm ausgehen. Von dieser Grundlage aus wollen wir uns über unser kleines Programm Gedanken machen.

Wie soll unsere kleine Programmierungsaufgabe aussehen?

Wir wollen zunächst einen Text, den wir per Programm im Computer gespeichert haben, auf dem Bildschirm sichtbar machen.

Dieser Text soll folgendermaßen lauten: "Willst du Basic lernen? Antworte bitte nur mit JA oder NEIN." Tippt jetzt der Benutzer "JA" ein, dann schreibt der Computer auf den Bildschirm: "Herzlichen Glückwunsch". Tippt er aber "NEIN" ein, dann schreibt der Computer: "Das ist aber schade". Sollte der Benutzer weder mit JA noch mit NEIN,

sondern mit irgend etwas anderem antworten, dann schreibt der Computer erneut: "Antworte bitte nur mit JA oder NEIN" und wartet dann auf die neue Antwort.

Dies soll unser kleines Dialogprogramm werden.

Ist ein Computer wirklich dialogfähig?

Der Dialog zwischen Mensch und Maschine kommt nur zustande, weil der Programmierer sich vorher den Dialog ausgedacht und als Programm in die Maschine eingegeben hat.

Der Computer selbst ist im Grunde seines Wesens nur ein Idiot, der eins und eins zusammenzählen kann. Dies tut er allerdings mit elektronischer Geschwindigkeit, ohne zu ermüden und beinahe fehlerfrei. Ein dialogfähiger Computer ist deshalb immer nur so geistreich wie der Mensch, der ihn programmiert hat. Diese Erkenntnis ist sehr wichtig und man sollte sie niemals vergessen, auch wenn es eines Tages von denkenden Schachcomputern und lernfähigen Robotern nur so wimmeln sollte.

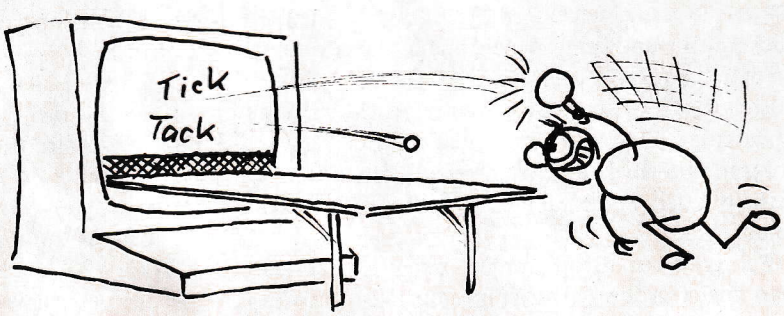
Wie bereiten wir den Dialog mit dem Computer vor?

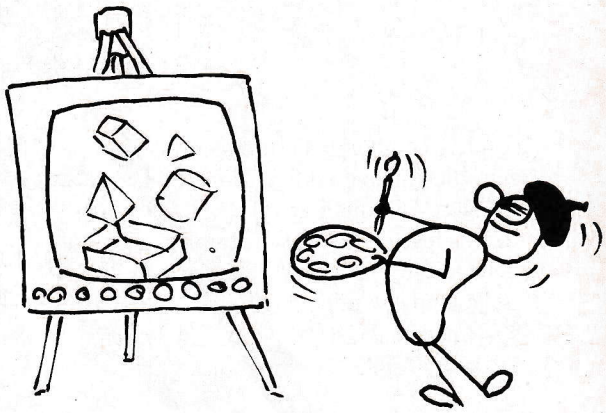
Der Dialog mit dem Computer findet mit Hilfe von Tastatur und Bildschirm statt, wobei wir unser besonderes Augenmerk auf den Bildschirm richten müssen.

Dies hat seinen Grund darin, daß nicht nur der Ausgabebefehl (wie z.B. PRINT A) etwas auf den Bildschirm schreibt, sondern auch der Tastatur-Eingabebefehl (wie z.B. INPUT B) das Eingetippte zur Kontrolle auf dem Bildschirm anzeigt.

Obwohl INPUT also ein Eingabebefehl ist, erfolgt zusätzlich noch eine Sofortausgabe auf dem Bildschirm. Dies ist eine sogenannte Echofunktion, weil die Entwickler der BASIC-Sprache es für nützlich hielten, daß der symbolische BASIC-Befehl INPUT nicht nur einen Tastaturein gabebefehl, sondern zusätzlich auch noch einen Bildschirmausgabebefehl erzeugt.

Wenn wir also jetzt im Zuge der Programmierungsvorbereitungen Datenträgermuster zeichnen, dann ist zwar das Bildschirmgerät die Ausgabeeinheit, aber der Bildschirm ist sowohl Ausgabedatenträger als auch Eingabedatenträger, da er auf Grund der Echofunktion sowohl Eingabe- als auch Ausgabedaten enthält. Von diesem Bildschirm und seinem Inhalt wollen wir uns jetzt ein Muster zeichnen.





Wie zeichnet man ein Bildschirmmuster

Um ein Bildschirmmuster zu zeichnen, muß man zunächst wissen, wie groß der Bildschirm des betreffenden Computers ist.

Während Businesscomputer und leistungsfähige Personalcomputer 80 Zeichen pro Zeile darstellen können, bringen die meisten Heimcomputer nur magere 32 oder 40 Zeichen pro Zeile auf den Bildschirm. Dies hat zum einen seinen Grund darin, daß die Heimcomputer als Bildschirmgeräte keine Monitore,

Computersysteme Monika Glitsch

64 K Speichererweiterung für Atari 600 XL
zum Einbau in das Gerät!

*

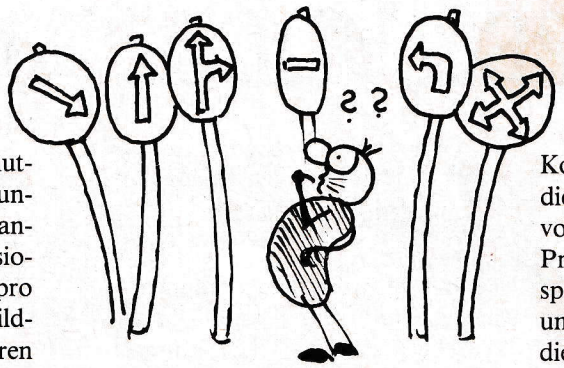
Für Spielfans:
TELELUPEN bringen eine unglaubliche Vergrößerung

*

Jegliches COMPUTER-ZUBEHÖR, DRUCKER, ARBEITSPLATZ-COMPUTER,
und, und, und ...

... und natürlich auch **SOFTWARE**

Tel.: 07152 / 4 76 43
Ludwig Finckh Weg 31/1
7250 Leonberg 1



sondern einfache Fernsehgeräte benutzen, die eine wesentlich gröbere und unruhigere Bildqualität ergeben. Zum anderen liegt es daran, daß ein professioneller Bildschirm mit 80 Zeichen pro Zeile und 24 oder 25 Zeilen pro Bildschirm einen viel größeren und teueren Bildschirmspeicher benötigt, als ein Bildschirm mit 22 Zeilen à 32 Zeichen.

Profis besitzen zum Zeichnen von Bildschirmmustern spezielle Vordrucke, die ein getreues Abbild des betreffenden Bildschirmes in Bezug auf Zeilen und Stellen darstellen. In diese Vordrucke kann man dann ganz einfach an den gewünschten Stellen zeilen- und stellengenau die leeren Masken oder bereits die beabsichtigten Texte einzeichnen. In Notfällen und für unsere Heimcomputer genügt allerdings auch einfaches kariertes Papier.

Wie soll unser Bildschirmentwurf aussehen?

Nachdem wir uns mit Bleistift, kariertem Papier und dem berühmten großen Radiergummi bewaffnet haben, schreiben wir jetzt in die erste Zeile ab Stelle 10: BASIC-PROGRAMM NR. 1

In die zweite Zeile ab Stelle 1 schreiben wir: EIN- UND AUSGABE-DIALOG. Darunter in die dritte Zeile ab Stelle 10: DATUM 30. APRIL 1984. Dann folgt in Zeile vier ab Stelle 10: MASCHINENTYP: ZX 81. Und nun beginnt in der fünften Zeile ab Stelle 10 eure Geburtsstunde als Programmierer. Ihr schreibt zunächst: PROGRAMMIERER: und tragt dann mit stolzgeschwellter Brust euren Namen ein.

Anschließend laßt ihr zwei Zeilen frei und tragt in die achte Zeile ab Spalte 1 ein: WILLT DU BASIC LERNEN? In Zeile neun kommt ab Stelle 1 die Aufforderung: ANTWORTE BITTE und in Zeile zehn ab Stelle 1 schreibt ihr noch: ABER NUR MIT JA ODER NEIN. Nun folgt in Zeile elf ab Stelle 1 ein leeres Kästchen von vier Stellen Länge. In dieses leere Kästchen wird dann später die Antwort JA oder NEIN eingegeben. Ihr könnt aber als Gedankenstütze schon jetzt in dieses Kästchen NEIN eintragen und dahinter in Klammern JA, damit jeder Betrachter sofort weiß, was dort eingegeben werden soll.

Danach lassen wir eine Leerzeile und zeichnen dann in Zeile 12 ebenfalls ab Stelle 1 ein leeres Kästchen von 23 Stellen Länge. Dahinein kommt der Text: HERZLICHEN GLUECKWUNSCH bzw. darunter in Klammern: DAS IST ABER SCHADE.

Dieses Bildschirmmuster ist der einzige Ein- und Ausgabe-Datenträgerentwurf, den wir für unser erstes Programm brauchen. Wir müssen uns jetzt nur noch Vorstellungen über den Programmablauf machen.

Was ist ein Programmablaufplan?

Ein Programmablaufplan ist eine graphische Darstellung, die einen Überblick gibt, was im Programm geschieht. Dies ist äußerst wichtig, weil bei größeren Programmen sonst viel zu leicht der Überblick verloren geht.

Man zeichnet dazu Kästchen besonderer Form und schreibt dort hinein, was im Programm geschieht. Außerdem muß man noch Richtungspfeile zwischen den einzelnen Kästchen anbringen, damit man sieht, in welcher Reihenfolge die Befehle im Programm tatsächlich ablaufen.

Man unterscheidet u.a. grobe Ablaufpläne, die nur einen allgemeinen Überblick geben, und feine Ablaufpläne, die so detailliert sind, daß praktisch jedes Kästchen schon einem Befehl entspricht, so daß man ohne Schwierigkeiten davon programmieren kann. Man nennt sie deshalb auch codierfähige Programmablaufpläne.

Bei größeren Programmen zeichnet man zuerst grobe Ablaufpläne, um das Konzept festzulegen und daraus dann die codierfähigen, die unter Umständen von Computer zu Computer oder von Programmiersprache zu Programmiersprache differieren. Wir zeichnen für unser kleines Programm sofort einen codierfähigen Programmablaufplan.

Was für Kästchen benutzt man für den Programmablaufplan?

Profis und solche, die es werden wollen, zeichnen ihre Kästchen gemäß den DIN-Symbolden.

Welche Kästchen man zeichnen muß, hängt von den verwendeten Befehlen ab. In unserer ersten Programmieraufgabe kommen bereits die wichtigsten Befehlsarten vor, und für jede dieser Befehlsarten gibt es spezielle Kästchen.

1. Ein- und Ausgabebefehle: Ein rechtsseitig verschobenes Parallelogramm. Z.B.:

INPUT
A

2. Logische Entscheidungen: Eine längliche Raute mit zwei Ausgängen. Der eine Ausgang ist für die Ja-Entscheidung, der andere Ausgang ist für die Nein-Entscheidung. Z.B.:

IF
A = B
GOTO

3. Allgemeine Verarbeitungsbefehle: Ein längliches Rechteck. Z.B.:

LET
A = B + C

4. Anfangs-, End- und Verbindungspunkte: Ein längliches Kästchen mit abgerundeten Ecken für die Anfangs- und Endpunkte. Z.B.:

ANFANG

Und ein kleiner Kreis mit der Numer des nachfolgenden Befehls für die Verbindungspunkte. Z.B.:

320

5. Richtungspfeile: Einfache Pfeile zwischen den einzelnen Kästchen. Z.B.:

↓ →

Diese Symbole benutzen wir von nun an, um unseren Programmablaufplan zu zeichnen. ■

C.W.

Atari-Zubehör aus USA

Wer einen Atari-Computer sein eigen nennt, kenn vielleicht auch den Compyshop. COMPUTER Praxis schaute, was sich hinter dieser Postadresse verbirgt.

Eine Überraschung ist der Compyshop in Siershahn im Westerwald. Denn wer einen Computerladen mit Ausstellungsraum und Schaufenster erwartet, der sucht hier vergeblich. Der 'Laden' befindet sich in einem Raum im Privathaus des Inhabers, Kundenberaters, Einkäufers und Versandleiters Wolfgang Kieckbusch.

Damit sind auch bereits einige der Besonderheiten des Compyshop aufgezählt. Wer hier einkauft, der bestellt seine Ware und bekommt sie per Post ins Haus geschickt. Zuvor wird der Kunde mit dem Wissen in allen Fachbereichen des Inhabers so umfassend beraten, daß er wenig Möglichkeiten hat, einen Fehlkauf zu tun. Hat er jedoch nach der Lieferung noch Fragen, so genügt meistens ein Anruf, um das Problem zu beheben.

Die Betreuung seiner Kunden ist das Wichtigste für Wolfgang Kieckbusch. Und darin hat er auch Erfahrung, denn bis 1982 war er Vertreter von Atari. Der Computer für zu Hause war damals immer noch ein unbekanntes Wesen, und es war unmöglich, in Koblenz einen Vertriebspartner für die Atari-Geräte zu finden — so ergriff Wolfgang Kieckbusch selber die Initiative: "Wir mieteten das Schaufenster einer Apotheke, stellten die Geräte da hinein und verkauften pro Woche zwei bis drei Stück." Der Test verlief erfolgreich, und so wurde der Compyshop im Herbst 1981 in Andernach eröffnet. Auf dem deutschen Markt gab es für die Computer, außer von Atari selber, überhaupt kein Zubehör. Der selbsternannte Einkäufer des neuen Computershops flog also nach Amerika. Dort wurde bereits Peripherie und Zubehör angeboten, von dem man auch heute noch in Deutschland träumen könnte, gäbe es nicht den Compyshop. Auch heute noch fliegt Wolfgang Kieckbusch zweimal jährlich in die USA, um dort die neueste Peripherie einzukaufen. Aus diesem Grunde bekommt man hier einfach alles. Im Sortiment befinden sich Speicherkarten aller Größen, Diskettenstationen, Tastaturen, Drucker- und RTTY-Interfaces, Staubschutzhauben und vieles mehr. Daß die neueste Software (z.B. BASIC XL, Action) in kürzester Zeit lieferbar ist, versteht sich von selbst. Doch auch wer älter oder ausgefallene Programme sucht, wird hier fündig. Der Atari Macroassembler, ein Modul zum Erstellen von Bildschirmausdrucken von Maschinenprogrammen sowie professionelle Finanzbuchhaltung sind am Lager. Weitere Neuheiten und Spezialperipherie sind weiter unten noch einmal aufgeführt.

Die Devise des Compyshop heißt: "Alles für den Atari muß da sein". Bei dieser Philosophie muß das Angebot natürlich straff gehalten werden. So wurde der Verkauf von Commodore Produkten gänzlich eingestellt. Erhältlich sind derzeit Atari-Geräte, SWP-Computer, die Spectravideo-Computer sowie der neue Genie-16, ein weitgehend IBM-kompatibler Rechner. ■ *M.A.*

Compyshop, Bergstr. 18, 5433 Siershahn,
Tel. 02623 / 66 76

Inhaber, Berater und Versandleiter Wolfgang Kieckbusch in seinem ‚Laden'.

Folgende Produkte sind uns im Compyshop aufgefallen:

Rana-Diskettenstationen: Das Double-Density Laufwerk für den Atari.

Digitalmikrofon: In das Mikrofon gesprochene Worte werden im Atari oder auf Diskette gespeichert und können in beliebigem Tempo wieder abgerufen werden. Die Qualität steht in gutem Verhältnis zum Preis.

Video-Interface: Das Interface digitalisiert das Signal einer Videokamera und bereitet es für die Speicherung im Atari auf. Sehr gut, aber leider nicht ganz billig.

Koala Pad: Dieses Zeichenbrett stellt mit der zugehörigen Software ein universelles Graphikprogramm für den Atari dar. Es können Bibliotheken ganzer Bilder oder einzelner Ausschnitte erzeugt werden. Alle Menüfunktionen sind vom Zeichenbrett aus anwählbar.

BASIC XL: Das neue BASIC von OSS, den Entwicklern des Atari OS, DOS und BASIC. Ein 24 K-Byte Interpreter, voll kompatibel zu Atari- und Microsoft-BASIC. Verfügt über einen Fast-Mode der besonders in Programmen mit vielen Sprüngen die Ausführungszeit erheblich verringert. Das Modul verwendet Bank-Select und läßt, in Verbindung mit dem ebenfalls neuen XL-DOS, 37 K Speicherplatz im Atari 800 frei.

ACTION!: Eine Compilersprache speziell für die Entwicklung von schnellen Spielprogrammen.

Die Sprache des Z80

Teil 2

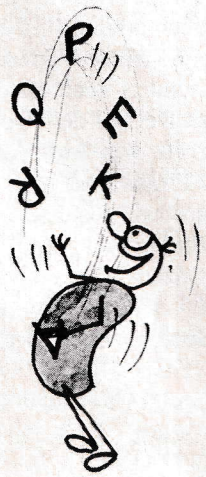
Im letzten Heft haben wir schon einiges über die Potenzen zur Basis 16 erfahren, aber ...
Wozu braucht man die Potenzen zur Basis 16?

Die 16er Potenzen benötigt man zum Ausrechnen von Speicherplatzadressen, für Kapazitätsberechnungen oder für Adressierungsprobleme. Da wir aber von klein auf das Dezimalsystem erlernt haben, wird selbst ein langjähriger EDV-Profi immer noch 'dezimal denken'. Entweder rechnet man eine vorhandene Hexadezimaladresse in eine Dezimaladresse um, um Größenvorstellungen zu bekommen. Oder man hat dezimale Rechenergebnisse, die man in eine maschinengerechte hexadezimale Form bringen muß.

Bevor wir uns aber mit leidigen Rechenkunststücken beschäftigen, wollen wir schnell noch zwei nützliche Tabellen kennenlernen. Die erste Tabelle ist sehr klein und für gute Rechner völlig ausreichend. Die zweite Tabelle ist für notorische Faulpelze gedacht, aber auch für Kontrollzwecke, denn oft ist man als Programmierer so übermüdet oder abgekämpft, daß man kaum eins und zwei richtig zusammenzählen kann — und dann dankt man Gott, wenn man ein zuverlässiges Hilfsmittel hat.

Tabelle der Potenzen zur Basis 16

$16^0 = 1$
$16^1 = 16$
$16^2 = 256$
$16^3 = 4096$
$16^4 = 65536$



Umrechnungstabelle für Hexadezimalzahlen

$n \cdot 16^3$			$n \cdot 16^2$			$n \cdot 16^1$			$n \cdot 16^0$		
H		D	H		D	H		D	H		D
0000	=	0000	000	=	000	00	=	00	0	=	0
1000	=	4096	100	=	256	10	=	16	1	=	1
2000	=	8192	200	=	512	20	=	32	2	=	2
3000	=	12288	300	=	768	30	=	48	3	=	3
4000	=	16384	400	=	1024	40	=	64	4	=	4
5000	=	20480	500	=	1280	50	=	80	5	=	5
6000	=	24576	600	=	1536	60	=	96	6	=	6
7000	=	28672	700	=	1792	70	=	112	7	=	7
8000	=	32768	800	=	2048	80	=	128	8	=	8
9000	=	36864	900	=	2304	90	=	144	9	=	9
A000	=	40960	A00	=	2560	A0	=	160	A	=	10
B000	=	45056	B00	=	2816	B0	=	176	B	=	11
C000	=	49152	C00	=	3072	C0	=	192	C	=	12
C000	=	53248	D00	=	3328	D0	=	208	D	=	13
E000	=	57344	E00	=	3584	E0	=	224	E	=	14
F000	=	61440	F00	=	3840	F0	=	240	F	=	15

$F000_H + F00_H + F0_H + F_H = FFFF_H$
$61440_D + 3840_D + 240_D + 15_D = 65535_D$

Wie setzt man Zahlen von einem Zahlensystem in das andere um?

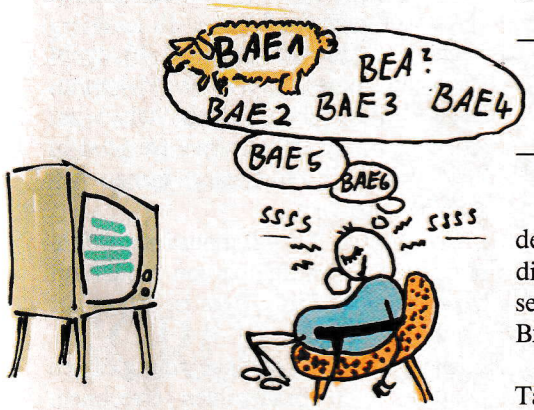
Um Zahlen von einem System in ein anderes umzuwandeln oder zu konvertieren, wie der Fachmann sagt, kann man drei Wege gehen:

1.) Man bedient sich eines dafür geeigneten Taschenrechners, der über eine HEX-DEC-Umwandlung verfügt. Dieser Weg ist der bequemste.

2.) Man vollzieht die Umwandlung im

Kopf. Dieser Weg ist der unsicherste.
3.) Man führt die Umwandlung mittels Bleistift und Papier durch. Dieser Weg ist wohl der üblichste, aber leider auch der mühsamste.

Für den letzten Weg gibt es zwei Methoden. Entweder bedient man sich besonderer Umrechnungstabellen, wie wir sie schon kennengelernt haben, oder wir benutzen spezielle Umrechnungsverfahren, die allerdings einige Grundkenntnisse über den Aufbau von Zahlensystemen voraussetzen. Da diese Kenntnisse uns auch später noch bei anderen Gelegenheiten von Nutzen sein werden, wollen wir schnell noch einen kleinen Ausflug in unsere Zahlensysteme vornehmen, dabei geht es besonders um den Stellenwert.



Was ist ein Stellenwertsystem?

Unsere heutigen modernen Zahlensysteme, wie z.B. das Dezimalsystem, beruhen auf dem Stellenwertsystem. Beim Dezimalsystem stehen auf der äußeren rechten Stelle die Einer, links davon die Zehner, wieder links davon die Hunderter, noch weiter links davon die Tausender usw. Bei der Zahl 3258_D handelt es sich um acht Einer (= 8 mal 10 hoch 0), fünf Zehner (= 5 mal 10 hoch 1), zwei Hunderter (= 2 mal 10 hoch 2) und drei Tausender (= 3 mal 10 hoch 3). Zählen wir dies alles zusammen, so erhalten wir 3000 plus 200 plus 50 plus 8, was, wie könnte es auch anders sein, genau unsere Zahl 3258_D ergibt.

Genauso ist auch das Hexadezimalsystem aufgebaut. Bei der Zahl $BE8F_H$ handelt es sich um fünfzehn Einer (= F mal 16 hoch 0), acht Sechzehner (= 8 mal 16 hoch 1), vierzehn Zweihundertsechsundfünfziger (E mal 16 hoch 2) und elf Viertausendsechsundneunziger (= B mal 16 hoch 3). Wenn wir wissen wollen, welcher Dezimalzahl die Zahl $BE8F_H$ entspricht, können wir dies leicht mit Hilfe unserer Hexadezimaltabellen ermitteln, indem wir die Dezimalwerte für $B000_H$, $E00_H$, 80_H und F_H heraussuchen und addieren. Also $45056_D + 3584_D + 128_D + 15_D = 48783$.

Jede Stelle besitzt eine bestimmte Wertigkeit, die sich aus der Basis des betreffenden Systems und der Potenz, die dieser Stelle zugehörig ist, ergibt.

Wie wandelt man Hexadezimalzahlen in Dezimalzahlen um?

Nehmen wir an, wir haben die Hexadezimalzahl CBA_H, dann können wir diese nach dem Stellenwertsystem auflösen in C00 + B0 + A, oder $Cx16^2$, plus $Bx16^1$, plus $Ax16^0$.

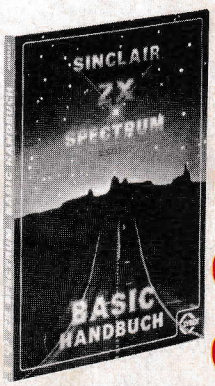
Jetzt schauen wir in unserer kleinen Tabelle der Potenzen zur Basis 16 nach und entdecken: $16^2 = 256$, $16^1 = 16$ und $16^0 = 1$ also können wir jetzt dafür auch schreiben: Cx256, + Bx16, + Ax1. Nun rechnen wir die einzelnen Produkte aus:

C mal 256 = 12 mal 256 = 3072
B mal 16 = 11 mal 16 = 176
A mal 1 = 10 mal 1 = 10
Dies ergibt 3072 + 176 + 10 = 3258_D

Wissen von Sybex:

SINCLAIR ZX SPECTRUM BASIC HANDBUCH

Für jeden SPECTRUM-Anwender stellt dieses Handbuch eine wichtige Hilfe dar. Es gibt eine Übersicht für alle BASIC-Begriffe, die auf diesem Rechner verwendet werden können. Anhand von Beispielen werden diese Begriffe ausführlich erläutert. Zahlreiche Programme dienen zugleich als Anregung für den Anwender, um fremde Programme besser verstehen und eigene leicht entwickeln zu können.



D. Hergert
SINCLAIR ZX SPECTRUM BASIC HANDBUCH
288 Seiten
188 Abbildungen
Ref.-Nr. **3027** (1983)
DM 32,–

Sprechen Sie die Sprache, die Ihr SPECTRUM versteht!

Sybex-Bücher sind erhältlich bei Ihrem Buch- und Fachhändler. Fragen Sie danach!

Verlagsauslieferung:
Österreich: Fachbuch-Center ERB, Amerlingstr. 1, 1061 Wien
Schweiz: Versandbuchhandlung Thali AG, Industriestr. 2, 6285 Hitzkirch, Tel. 041/852828

Direktbestellungen beim Verlag gegen Verrechnungsscheck (+DM 2,50 Versandkostenanteil)

Fordern Sie ein Gesamt-Buch-Verzeichnis an.

Besuchen Sie uns auf der Internationalen Computer Show, 14.-17. Juni in Köln. Halle 13 OG, Stand D5

SYBEX

SYBEX-VERLAG GMBH
Postfach 30 09 61
4000 DÜSSELDORF 30
Tel. 0211-626441, Telex: 8588163

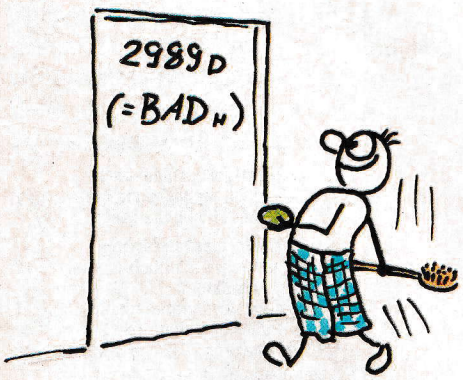
Also lautet die Hexadezimalzahl CBA_H als Dezimalzahl 3258_D.

Noch schneller geht es natürlich unter Verwendung der großen Tabelle. Wir finden dort: C00 = 3072, B0 = 176 und A = 10. Also
$3072 + 176 + 10 = 3258_D$.

Wie werden Dezimalzahlen in Hexadezimalzahlen per Tabelle umgewandelt?

Angenommen, wir haben die Dezimalzahl 3258_D und wollen diese in eine Hexadezimalzahl umwandeln, dann können wir dies sowohl mit unserer großen Tabelle als auch mit Hilfe eines besonderen, klugen Umrechnungsverfahrens durchführen.

Zunächst die Umwandlung per Tabelle: Wir ermitteln aus der Tabelle denjenigen Wert, der am dichtesten unter unserer Zahl liegt. Dies ist laut Tabelle die Zahl $3072_D = C00_H$. Dann ziehen wir diesen Wert von 3258 ab. Das ergibt einen Rest von 186_D. Nun suchen wir wieder in der Tabelle nach und finden als nächstniedrigen Wert $176_D = B0_H$.



Diese 176_D zeihen wir von 186_D ab und erhalten einen Rest von $10_D = A_H$. Die Addition ergibt nun $C00_H + B0_H + A_H = CBA_H$. Also ist $3258_D = CBA_H$. Oder in Stellenwertpositionen untereinander geschrieben:

C00 = 3082
B0 = 176
A = 10

$CBA_H = 3258_D$

Wie werden Dezimalzahlen in Hexadezimalzahlen durch ein besonderes Rechenverfahren umgewandelt?

Nehmen wir noch einmal unsere Dezimalzahl 3258_D, und versuchen wir jetzt, die Umrechnung ohne Tabelle vorzunehmen. Dies können wir erreichen, indem wir die Zahl gewissermaßen von hinten her aufdröseln. Wir erinnern uns dabei, daß das Hexadezimalsystem nach dem gleichen Prinzip aufgebaut ist, wie das Dezimalsystem, also ganz rechts die Einer, dann die 16er, dann die 256er, dann die 4096er usw.

Wir teilen dann zuerst 2358_D durch 16_D, dann erhalten wir 203_D, Rest 10. Dieser Rest sind die Einer im Hexadezimalsystem. Da 10_D gleich A_H ist, schreiben wir also A hin.

Dann teilen wir die erhaltenen 203_D durch 16_D und erhalten 12_D und einen Rest von 11_D. Dieser Rest sind die 16er für das Hexadezimalsystem. Da 11_D gleich B_H ist, schreiben wir also B hin.

Übrig bleibt 12_D. Dies sind die 256er für das Hexadezimalsystem. Da 12_D gleich C_H, schreiben wir das C noch links vor das B. Somit erhalten wir die Zahl CBA_H, die, falls wir richtig gerechnet haben, gleich der Zahl 3258_D sein muß. Wir machen deshalb zur Sicherheit die Probe:

C00 = 3072
B0 = 176
A = 10

$CBA_H = 3258_D$

Wie kommt die Umrechnung zustande?

Die Umrechnungsmethode ist eine Umkehrung der Stellenwertaddition. Dies kann man am leichtesten am Dezimalsystem erklären, obwohl es dort zugegebenermaßen trivial ist.

Wie wir wissen, kann man die Zahl 3258_D auch als Addition der Stellenwerte nämlich 3000 + 200 + 50 + 8 schreiben. Dies ergibt dann 3258. Teilt man nun 3258 durch 10, so erhält man 325 Rest 8. Die 8 sind die Einer. Nun teilt man 325 durch 10 und erhält 32 Rest 5. Diese 5 sind die Zehner. Sodann teilt man 32 durch 10 und erhält 3 Rest 2. Dieser Rest 2 sind die Hunderter. Die verbleibenden 3 sind die Tausender. Also $3000 + 200 + 50 + 8 = 3258_D$, was auch als Ergebnis schließlich auch herauskommen muß.

Das ständige Teilen durch 10 ergibt Reste mit immer höheren Stellenwerten. Nach dem gleichen Prinzip haben wir auch die Umwandlung von Dezimal in Hexadezimal im vorhergegangenen Abschnitt vollzogen.

Es ist eine Art "Exhaustionsverfahren", eine Aufräufelung von hinten auf hexadezimaler Basis. Dies ist schon der ganze Trick.

Übrigens lassen sich nach der gleichen Melodie auch Binärzahlen errechnen. Man muß die Dezimalsache nur fleißig durch 2 teilen und dann die jeweiligen Reste hinschreiben.

Als Belohnung für eure Aufmerksamkeit werden wir das nächste Mal die Architektur der Z80 CPU erläutern. ■

C. W.

Verständigungsschwierigkeiten

Was der Spectrum mit UNPLOT anfangen kann

Eines der zwei Hauptprobleme, die sich beim Übertragen von ZX81 Programmen auf den großen Bruder ZX Spectrum ergeben, ist die Umsetzung von UNPLOT, das des öfteren in 81er Listings Verwendung findet.

Die Transformation dieses Statements ist sehr einfach, da der Spectrum über den gleichen Befehl verfügt, dieser jedoch sinnigerweise anders bezeichnet ist. Statt des Befehles UNPLOT hat der durch die Informationspolitik von Sinclair gebeutelte User nur den Befehl PLOT OVER einzugeben, und das ZX81 Listing läuft auch auf seiner Maschine.

Es gibt für den ZX81 mehrere Programme, die mit Trick arbeiten, um das Programm auch für den kleineren Speicherbereich noch lauffähig zu erhalten. Hier ist Vorsicht geboten. Steht zum Beispiel vor einem ZX81 Programm, daß vor dem Starten bestimmte Anweisungen im Direktmodus einzugeben sind, ist es für den Spectrum-User sinnvoll, diese Programmzeilen gleich richtig im Programm zu verankern. Man schreibt die Direktanweisungen dann normalerweise in die Zeilen von eins aufwärts. Steht in einer der benötigten Zeilennumern bereits etwas, kann man, wenn es sich nicht um einen GOTO oder GOSUB handelt, einfach in der nächsten Zeile weiterschreiben. Ansonsten muß die Sprunganweisung hinter die 'Direktzeilen' verschoben werden. Hat man zum Beispiel in Zeile 1 einen Sprungbefehl und muß vier 'Direktzeilen' eingeben, schreibt man den Sprungbefehl einfach in Zeile 5 und die Direktzeilen in die Zeilen 1—4.

Die Programme können dann auch entgegen der Anweisung vom ursprünglichen Begleittext zu dem Listing mit RUN statt mit GOTO gestartet werden. ■

G.K.

SCROLL auf dem ZX Spectrum

Grundsätzlich kann man jedes Programm, das auf dem ZX 81 geschrieben ist, auch für den ZX Spectrum gangbar machen, da das Spectrum-BASIC wesentlich komfortabler ist und die meisten Befehle des ZX 81 ohnehin beinhaltet. Ab und zu stößt man als ZX Spectrum Besitzer trotzdem auf Befehle, die einem unbekannt sind und daher eine Umsetzung erforderlich machen. Für einen dieser Befehle, den Befehle SCROLL, möchten wir hier Stelle die Übersetzung vorstellen.

Das sehr komfortable Betriebssystem des Spectrums kommt uns bei diesem Unterfangen zu Hilfe. Wie jeder Spectrumbenutzer, der schon einmal ein Programm gelistet hat, weiß, scrollt der Spectrum das Listing von unten nach oben über den Bildschirm und fragt, nachdem eine Bildschirmseite vollgeschrieben hat: SCROLL? Diese Scrollroutine liegt im Betriebssystem bei #3180.

Hat man also ein ZX 81 Listing, in dem der Befehl SCROLL auftaucht, muß man als Spectrumbesitzer statt dieses Befehls Z = USR 3180 eingeben. Um zu verhindern, daß der Computer an dieser Stelle SCROLL? fragt, muß vorher die zur Kontrolle dieses Vorgangs benutzte Systemvariable entsprechend hoch gesetzt werden. Diese befindet sich im Speicher an der Stelle #23692. Der Wert, der sich in dieser Speicherstelle befindet, gibt an, nach dem wievielten SCROLL unterbrochen werden soll.

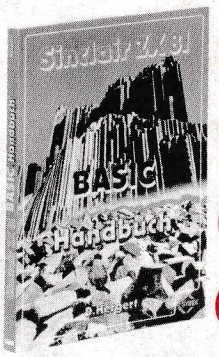
Als Ersatz für den SCROLL-Befehl im ZX 81 Listing könnte in der Spectrum Version also stehen:
POKE 23692, 255:Z = USR 3180 ■

G.K.

Wissen von Sybex:

SINCLAIR ZX81 BASIC Handbuch

Ihr ZX81 spricht BASIC. Sprechen Sie seine Sprache! Dieses Buch hilft Ihnen, alle Möglichkeiten Ihres ZX81 zu beherrschen. Das vollständige BASIC Vokabular wird anhand von praktischen Beispielen erläutert und beschrieben. Anhand der vielen Programmbeispiele lernen Sie das Programmieren mit Ihrem ZX81.



D. Hergert
SINCLAIR ZX81 BASIC Handbuch
184 Seiten, 120 Abb.
Ref.-Nr. **3028** (1983)
DM 28,–

Sprechen Sie die Sprache, die Ihr ZX81 versteht!

Sybex-Bücher sind erhältlich bei Ihrem Buch- und Fachhändler. Fragen Sie danach!

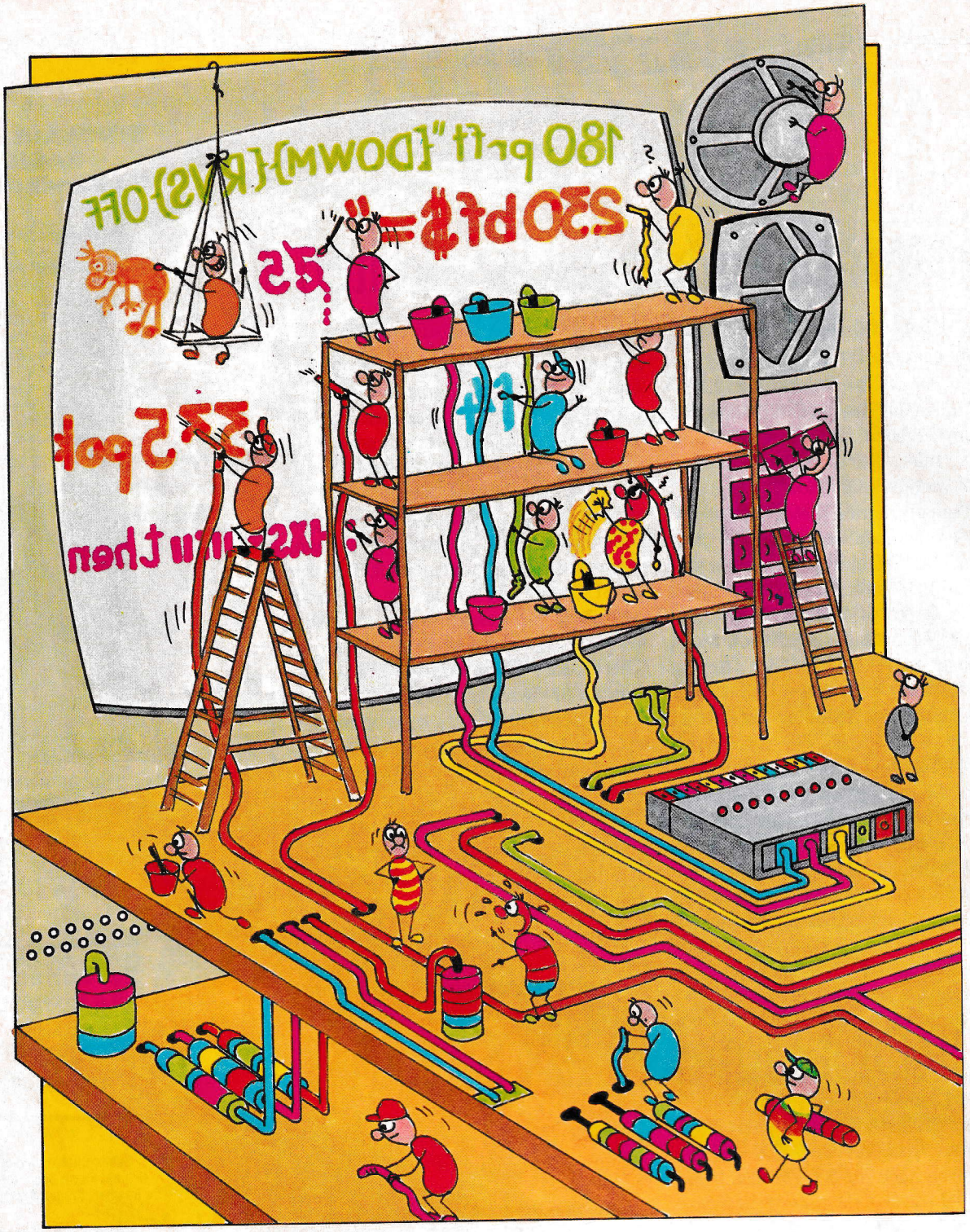
Verlagsauslieferung:
Österreich: Fachbuch-Center ERB, Amerlingstr. 1, 1061 Wien
Schweiz: Versandbuchhandlung Thali AG, Industriestr. 2, 6285 Hitzkirch, Tel. 041/852828
Direktbestellungen beim Verlag gegen Verrechnungsscheck (+DM 2,50 Versandkostenanteil)

Fordern Sie ein Gesamt-Buch-Verzeichnis an.

Besuchen Sie uns auf der Internationalen Computer Show, 14.-17. Juni in Köln. Halle 13 OG, Stand D5

SYBEX

SYBEX-VERLAG GMBH
Postfach 30 09 61
4000 DÜSSELDORF 30
Tel. 0211-626441, Telex: 8588163



COMPUTER Praxis-Vorzugskarte 6/84

An diese Adresse schicken Sie bitte Computer-Praxis zum ermäßigten Abonnementspreis von DM 48,– für 12 Hefte, das sind DM 4,– pro Heft, statt DM 5,– bei Einzelbezug. (20 % Preis-Vorteil)

Die Zustellgebühr trägt der Verlag. Die Lieferung beginnt mit der nächsterreichbaren Ausgabe.

Vorname Name

Straße/Nr.

PLZ Wohnort

Gewünschte Zahlungsweise bitte ankreuzen:

☐ **Bargeldlos durch Bankeinzug**

Kontonummer Name der Bank

Bankleitzahl

☐ gegen Rechnung (12 Hefte DM 48,–)
keine Vorauszahlung leisten. Rechnung abwarten.

Wichtiger Hinweis:
Sie garantieren mir, daß ich diese Vereinbarungen innerhalb einer Woche widerrufen kann.

Datum

Unterschrift

Auslandspreis Europa DM 57,–.

Monatlich zum Abo-Vorteil

- 12 Hefte zum ermäßigten Abonnementspreis von DM 48,–, das sind DM 4,– pro Heft statt DM 5,–. (20 % Preis-Vorteil)
- **Als Geschenk das Poster** (vierfarbig DIN A3)
- Lieferung per Post frei Haus
- Keine Kündigungsfrist. Sie können jederzeit auf die Lieferung wieder verzichten. Bereits bezahlte Gebühren für noch nicht gelieferte Hefte erhalten Sie zurück.

COMPUTER Praxis
Leserservice
Postfach 10 64 05
2000 Hamburg 11

Hier spricht Sinclairs Spectrum

Welche Taste ihr auch drückt: Euer Computer spricht's aus. Sein Sprachrohr heißt MicroSPEECH.

Eine der seltenen Neuheiten, die wir auf der Hobby-tronic '84 in Dortmund fanden, war der CURRAH MicroSPEECH — ein kleiner, schwarzer Kasten, der dem Sinclair-Spectrum eine Stimme gibt. Die Firma Goliath Electronics hat COMPUTER Praxis ein Exemplar zu Testzwecken zur Verfügung gestellt.

Der Anschluß des Gerätes an den Spectrum bereitet keine Schwierigkeit, und wer zu vorsichtig ist, den Micro-SPEECH einfach anzustöpseln, findet guten Rat in der, noch englischen, Betriebsanleitung. Neben Anweisungen zum Anschluß und für die Inbetriebnahme ist auch eine 'Laut-Liste' gegeben — die wohl wichtigste Liste überhaupt.

Wir haben versucht, ein kleines Programm zu schreiben:

```
10 FOR i =TO 1000:NEXT i
20 LET a$="hu(ll)(oo)"
30 LET b$="fr(oy)nde fon"
40 LET c$="Computer praksis"
50 LET s$=a$:PAUSE 1
60 LET s$=b$:PAUSE 1
70 LET s$=c$:PAUSE 1
```

Dabei haben wir uns, wie ihr wohl schnell erkannt habt, an die 'Laut-Liste' gehalten. Der MikroSPEECH kann nämlich eigentlich nur Englisch. Entsprechend klingt sein Deutsch: Stellt euch eine texanische Ente mit Schnupfen vor. Das kommt ungefähr hin. Man braucht schon eine Gewöhnungszeit bevor man die Mitteilungen seines Spectrums versteht.

Für manche Spiele mag es ganz witzig sein, sich die Meldungen nicht schriftlich sondern akustisch geben zu lassen. Aber uns nervte der Sound der Kommentare schon nach 10 Minuten, mit dem der MicroSPEECH jeden Tastendruck quittiert. Schade ist es auch, daß das Bild mit MicroSPEECH noch schlechter wird, als es sowieso schon ist!

Wesentlich sinnvoller wäre es, endlich ein brauchbares und auch kaufbares Interface auf den Markt zu bringen. RS 232 C, Centronics und paralleler Disketten-Port sind nützlicher und dringender als Meldungen in schlechtem Deutsch. ■

H.U.

ZX-Spectrum-Centipede

Dem Tausendfüßler auf die Füße getreten

Das ZX-Spectrum-Centipede ähnelt beiden bekannten Spielen 'Centipede' und 'Space-Invaders'. Ziel ist es, die angreifenden Space Invaders abzuwehren. Gesteuert wird mit folgenden Tasten:

6 für links
7 für rechts
0 für Feuer.

Das Programm besteht aus zwei Teilen, einem BASIC- und einem Maschinenprogramm. Um die Eingabe zu erleichtern, befindet sich das Maschinenprogramm in den DATA-Zeilen eines zweiten BASIC-Programms. Zunächst wird das erste BASIC-Programm eingegeben und mit SAVE "Centipede" LINE 1 auf Cassette aufgenommen. Danach gibt man das zweite Programm ein und startet es mit RUN. Entdeckt das Programm eine fehlerhafte DATA-Zeile, so unterbricht es sich. Ist das Programm korrekt durchlaufen, dann befindet sich das Maschinenprogramm nun ab Adresse 31290 im RAM-Speicher. Es wird mit SAVE "Centipede" CODE 31290, 1470 direkt hinter dem ersten BASIC-Programm aufgenommen. Das Hilfsprogramm wird nun nicht mehr benötigt. Um das Spiel zu laden, braucht ganz einfach nur LOAD "" eingegeben zu werden.

16 K oder 48 K

Das Programm ist für den 16 K-Spectrum geschrieben. Die Besitzer des 48 K-Spectrum müssen eine zusätzliche Programmzeile in das erste BASIC-Programm einfügen, damit die USR-Graphik im Speicher an die entsprechende Stelle gesetzt wird:

41 POKE 23675,88: POKE 23676,127

Im Maschinenprogramm wird eine Routine verwendet, die der Funktion ATTR im BASIC entspricht. Sie dient zur Feststellung der Farben auf dem Bildschirm. Diese Routine läßt sich sehr gut in andere Maschinenprogramme einbauen und kann dabei an einer beliebigen Stelle im RAM-Speicher stehen. Sie wird mit einem CALL-Befehl aufgerufen. Vorher muß jedoch das HL-Registerpaar mit den Koordinaten der abzufragenden Zeichen-Position geladen werden (H: Spalte, L: Zeile). Nach Aufruf des Unterprogramms steht im Akku eine Zahl, die wie bei ATTR den Farben-Status der entsprechenden Zeichen-Postition angibt. Ihr Wert ergibt sich nach Tabelle 2.

Die restlichen Register bleiben unverändert, sie brauchen also nicht auf dem Stack gesichert zu werden.

Tabelle 1

Das BASIC-Programm ist folgendermaßen gegliedert:

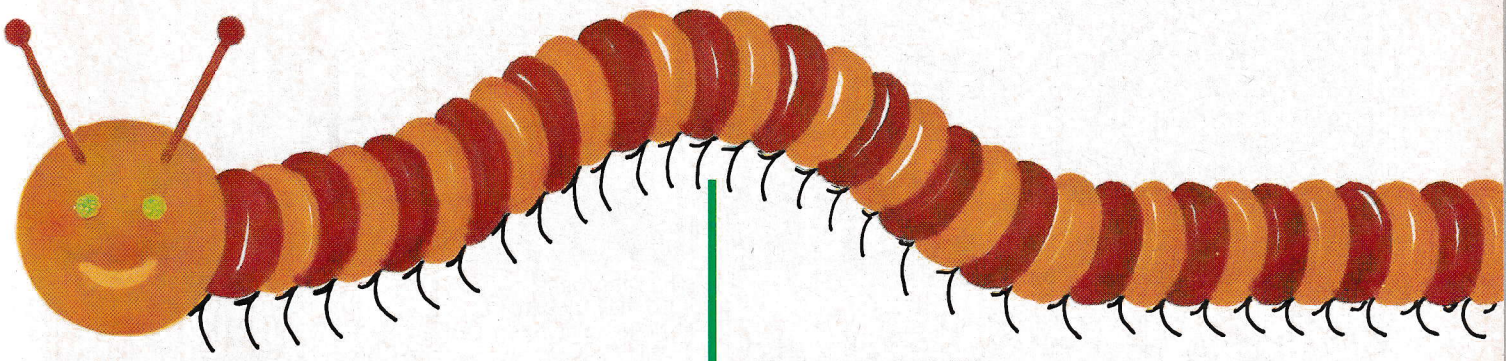
540	Aufruf des Maschinenprogramms
1000—2000	Verzweigung hierhin bei Beendigung einer Runde
2000—3000	Verzweigung hierhin, wenn man getroffen wurde
3000—4000, 7000—8000	USR-Graphik
4000—5000	Bildaufbau
5000—6000	Vorbereitungen für das Maschinenprogramm
6000—7000	Variablen

Tabelle 2

	128	bei FLASH 1
+	0	bei FLASH 0
+	64	bei BRIGHT 1
+	0	bei BRIGHT 0
+	8*	der Code der PAPER-Farbe
+		der Code der INC-Farbe
=		Farbenstatus der entspr. Zeichen-Position

Tabelle 3 ATTR-Maschinenroutine, entspricht Block 10

PUSH BC	C5	
PUSH HL	E5	Sichern der HL, BC Register
LD	H,0	26 00
ADD HL, HL	29	Ermittlung der Adresse im Farb-Speicher
ADD HL, HL	29	
ADD HL, HL	29	
ADD HL, HL	29	
ADD HL, HL	29	Y-Koordinate mal 32
POP BC	C1	
PUSH BC	C5	
LD C, B	48	
LD B, 0	06 00	
ADD HL, BC	09	+ X-Koordinate
LD BC, 22528	01 00 58	
LD A, (HL)	7E	Akku mit dem Wert dieser Adresse laden
POP HL	E1	
POP BC	C1	
RET		Rücksprung



```
  20 REM Copyright Jens Fretwurst
  40 CLEAR 31289: LOAD "CODE"
  70 LET HSCORE=0
 100 GO SUB 6000
 110 GO SUB 3000
 120 GO SUB 8000
 500 REM
 510 GO SUB 5000
 520 GO SUB 7000
 530 GO SUB 4000
 540 LET Z=USR 31354
 550 IF Z>5000 THEN GO SUB 2000: GO TO 570
 560 GO SUB 1000
 570 GO TO 500
1000 REM
1010 LET Z=USR 31290
1020 CLS:PAUSE 50
1030 PRINT INK 2; FLASH1; AT 8,8; " {INV.ON} {INV.OFF}
     {INV.ON} {INV.OFF} {INV.ON} {INV.OFF} {INV.ON}
     {INV.OFF} {INV.ON} {INV.OFF} {INV.ON} {INV.OFF}
     {INV.ON} {INV.OFF} "
1040 PRINT INK 2; FLASH1; AT 9,8; "{INV.ON} {INV.OFF}";
     AT 9,22;"{INV.ON} {INV.OFF}"
1050 PRINT INK 2; FLASH1; AT 10,7; "{INV.ON}
     {INV.OFF} "; AT 10,22; " {INV.ON} {INV.OFF}"
1060 PRINT INK 2; FLASH1; AT 11,8; "{INV.ON}
     {INV.OFF}"; AT 11,22; "{INV.ON} {INV.OFF}"
1070 PRINT INK 2; FLASH1; AT 12,8; " {INV.ON} {INV.OFF}
     {INV.ON} {INV.OFF} {INV.ON} {INV.OFF} {INV.ON}
     {INV.OFF} {INV.ON} {INV.OFF} {INV.ON} {INV.OFF}
     {INV.ON} {INV.OFF} "
1090 PRINT AT 10,10; "STORE:"; SCORE
1100 PAUSE 50: LET SC=(110-LV1)*10+LV2*10+
     (L-SPEED)*100+(LV2-2-PEEK 23673)*50
1110 FOR I=SCORE TO SCORE+SC STEP 10-20*(SC<0)
1120 PRINT AT 10,16; I: BEEP .02,40
1130 IF I/5000=INT (I/5000) AND I<>0 THEN LET S=S+1:
     PRINT AT 16,12; "BONUS F"
1140 NEXT I: LET SCORE = SCORE+SC
1150 LET LV1=LV1-10*(LV1>10)
1160 LET LV2=LV2+1*(LV2<16)
1170 LET NN=LV2: LET SP=SP+1
1180 IF SP=3 THEN LET SP=0: LET SPEED=SPEED-1*(SPEED>1)
1190 RETURN
2000 REM
2010 LET Z=USR 31310
2020 IF S=0 THEN GO TO 2500
2030 LET S=S-1
2040 LET NN=INT (LV2/2)+1
2050 RETURN
2500 REM
2510 IF SCORE>HSCORE THEN GO TO 2700
2520 FOR I=40 TO 10 STEP-1: BEEP .1,I: NEXT I
2530 PAUSE 300: GO TO 100
2700 IF HSCORE=0 THEN LET HSCORE=SCORE: GO TO 2530
2710 PAUSE 100: FOR I=1 TO 3
2720 LET Z=USR 31290: NEXT I
2730 CLS
2740 PRINT FLASH1; AT 4,5; "NEW HIGH-SCORE:"; SCORE
2750 LET HSCORE=SCORE
2760 PAUSE 300: GO TO 50
```

```
3000 REM USR-GRAPHIK
3010 RESTORE 3100: FOR I=32600 TO 32647
3020 READ PP: POKE I,PP
3030 NEXT I: RETURN
3100 DATA 0,0,0,0,0,0,0,0
3105 DATA 60,90,126,60,24,36,90,0
3120 DATA 28,28,126,126,52,52,20,12
3130 DATA 0,28,98,129,145,110,0,0
3140 DATA 16,56,56,56,56,56,124,124
3150 DATA 0,24,24,126,255,255,255,255
4000 RANDOMIZE PEEK 23672: CLS
4010 PRINT INK 7; AT 1,0; " {INV.ON} {INV.OFF}";
     AT 1,31; "{INV.ON} {INV.OFF}"
4020 FOR I=2 TO 21
4030 PRINT INK 7; AT I, 0; "{INV.ON} {INV.OFF}";
     AT I,31; "{INV.ON} {INV.OFF}"
4040 PRINT INK 3; AT I*(I<19),INT (RND*30)+1;"D"
4050 NEXT I
4060 FOR I=0 TO 31: PRINT #1; INK 7; AT 0,I; "{INV.ON}
```


Computer Zubehör

H. Hofmann & W.-D. Luther

Schützen Sie Ihre wertvolle Elektronik mit

● **Staubschutzhauben**

Legen Sie Ihre gelochten 3-Loch-Materialien ab in

● **3-Loch-Ordner** *(US-Maße)*

Nutzen Sie Ihre Disketten-Rückseite mit unserem

● **Disketten-Umdreh-Set**

Frischen Sie Ihre verbrauchten Farbbandkassetten mit schwarzer Farbbandfarbe auf mit dem

● **REFRESHER**

Bedrucken Sie Ihr chamois-getöntes Briefpapier in braun

● **Farbige Farbbandcassetten**

liefern wir für fast alle Drucker – auch Nadeldrucker – in den Farben braun, grün, rot und blau

Prospekt RR5 *gegen Freiumschlag:*

ComZu · H. Hofmann & W.-D. Luther

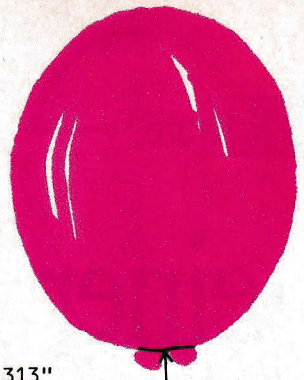
Ulmenweg 3 · 6086 RIEDSTADT 3

Händleranfragen willkommen!

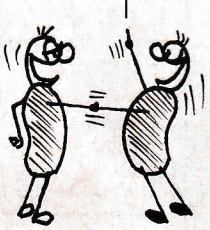
```
     {INV.OFF}": NEXT I
4070 PRINT INK 1; AT 21,16; "F"
4080 FOR I=1 TO S
4090 PRINT #1; INK 3; AT 1,I+22; "F"
4100 NEXT I
4110 PRINT £1; INK 3; AT 0,0; "SCORE:"; SCORE; AT 1,0;
     "HIGH-SCORE:"; HSCORE
4130 FOR I=1 TO PEEK 32654
4140 PRINT INK 2; AT 1, PEEK 32506+I-1; "B"
4150 NEXT I
4160 PAUSE 100
5000 REM
5010 POKE 32497,0
5020 POKE 32498,238: POKE 32499,126
5030 LET A=INT (RND*12)+2
5040 FOR I=32505 TO 32591 STEP 5
5050 LET A=A+1: POKE I,1
5060 POKE I+1,A: POKE I+2,0
5070 POKE I+3,0: POKE I+4,0
5080 NEXT I
5090 POKE 23673,0: POKE 23672,INT (RND*256)
5100 RANDOMIZE 100
5110 FOR I=32660 TO 32760: POKE I,INT (RND*LV1):
     NEXT I
5120 POKE 32648,148: POKE 32649,127
5130 POKE 32650,16
5140 POKE 32653,0: POKE 32654,NN
5150 POKE 31448,0: POKE 31449,SPEED
5160 POKE 31761,NN
5170 POKE 31770,NN*5
5180 LET I=32500+5*NN
5190 POKE 31856,I-256*INT (I/256)
5200 POKE 31857,INT (I/256)
5210 RETURN
6000 REM
6010 INK 0: PAPER 7: CLS
6020 LET LV1=100
6030 LET LV2=10
6040 LET SCORE=0
6060 LET NN=LV2
6070 LET S=3: LET SP=0
6080 LET SPEED=10
6090 RETURN
7000 RESTORE 7100+SP*10
7010 FOR I=32608 TO 32615
7020 READ PP: POKE I,PP
7030 NEXT I: RETURN
7100 DATA 24,60,90,126,24,36,90,0
7110 DATA 60,90,126,60,24,36,90,0
7120 DATA 195,60,126,90,255,102,90,129
8000 CLS
8100 LET A$="CENTIPEDE"
8110 LET Z=USR 31310
8120 FOR I=1 TO 9
8130 FOR N=19 TO 10 STEP-1: BEEP.01,40
8140 PRINT AT N,I+10; A$(I)
8150 PRINT AT N+1,I+10; " "
8160 NEXT N: NEXT I
8170 PAUSE 50
8180 PRINT AT 8,9; FLASH1; "{GRAPHICS}
     6666666666666"
8190 FOR I=9 TO 11: PRINT FLASH1; AT I,9;
     "{GRAPHICS}6"; AT I,21; "{GRAPHICS}6": NEXT I
8200 PRINT AT 12,9; FLASH1; "{GRAPHICS}66666666666666"
8210 PAUSE 100
8220 PRINT AT 21,0; "PRESS ANY KEY TO START THE GAME"
8270 FOR I=11 TO 19
8280 IF INKEY$<>"" THEN RETURN
8290 PRINT AT 10,I; OVER 1; "{INV.ON} {INV.OFF}"
8300 NEXT I: GO TO 8270
```

BASIC-LOADER
für Maschinenteil

```
  5 RESTORE
 10 FOR I=1 TO 11
 20 READ A,B,C: LET S=0: FOR N=A TO A+B-1 STEP 10
 30 READ A$: FOR M=1 TO LEN A$-1 STEP 2
 40 LET P=16*(CODE A$(M)-48*(CODE A$(M)<58)-55*
    (CODE A$(M)>64))
 50 LET P=P+(CODE A$(M+1)-48*(CODE A$(M+1)<58)-
    55*(CODE A$(M+1)>64))
 60 POKE (M-1)/2+N, P: LET S=S+P
 70 NEXT M: NEXT N
 80 IF S<>C THEN PRINT "ERROR IN BLOCK"; I: STOP
 90 NEXT I: PRINT "READY": STOP
 99 REM BLOCK 1
100 DATA 31290,16,1085
110 DATA "3E00010004110800DD21"
120 DATA "0800CDC87DC9"
129 REM BLOCK 2
130 DATA 31310,31,2024
140 DATA "3E00010018112000DD21"
150 DATA "0100CDC87D3E01010004"
160 DATA "112000DD210100CDC87D"
170 DATA "C9"
179 REM BLOCK 3
180 DATA 31354,86,9535
190 DATA "2A785C46CB38CB383A88"
200 DATA "7F803003066C9032887F"
210 DATA "CD107CCD487B2AD87A01"
220 DATA "0100ED4230F9CDDA7ACD"
230 DATA "487B2AD87A010100ED42"
240 DATA "30F9CD887CCD487B2AD8"
250 DATA "7A010100ED4230F9CDDA"
260 DATA "7ACD487B2AD87A010100"
270 DATA "ED4230F918AA"
279 REM BLOCK 4
280 DATA 31450,98,10239
290 DATA "2A897F3EEFDBFEFEEF20"
300 DATA "0125FEF7200124F52E15"
310 DATA "CD0E7EFE3828052A897F"
320 DATA "2E15E50E382A897F2E15"
330 DATA "11587FCD2C7EE17C328A"
340 DATA "7F0E3911807FCD2C7EF1"
350 DATA "FEFEC03A8D7FFE00C02A"
360 DATA "897F2E15228B7F3E0132"
370 DATA "8D7F3E010180021110 00"
380 DATA "DD210100CDC87DC9"
389 REM BLOCK 5
390 DATA 31560,200,16025
400 DATA "3A8D7FFE00C80E381158"
410 DATA "7FD52A8B7FE5CD2C7EE1"
420 DATA "D12D3EFFBD200C2C3E00"
430 DATA "328D7F0E38CD2C7EC922"
440 DATA "8B7FCD0E7EFE3A002842"
450 DATA "FE3B20070E3C11707F18"
460 DATA "1FFE3C20070E3E11707F"
470 DATA "1814FE3E20070E381158"
480 DATA "7F18090E3D11787FCD2C"
490 DATA "7EC9CD2C7E3E00328D7F"
500 DATA "3E01010002111800DD21"
510 DATA "0100CDC87DC90E381158"
520 DATA "7FCD2C7E3E0001000311"
530 DATA "1000DD210100CDC87D3A"
540 DATA "8E7F3D328E7FFE002004"
550 DATA "E10601C93E00328D7F2A"
560 DATA "8B7F11F97E06101ABD20"
```



```
570 DATA "16131A1BBC2010131313"
580 DATA "131AFE00280313180 93E"
590 DATA "0112C9131313131310DF"
599 REM BLOCK 6
600 DATA 31760,110,12449
610 DATA "060BC511F47E21F97E01"
620 DATA "3700EDB03AF87EFE0128"
630 DATA "4A2A887FC1C548060009"
640 DATA "7EFE01202E3AF17EFE2D"
650 DATA "28273C32F17E2AF27EED"
660 DATA "73F27EF92AF47EE52AF2"
670 DATA "7EED73F27EF901000111"
680 DATA "10003E00DD210400CDC8"
690 DATA "7D3AF47EFE152004C1C1"
700 DATA "47C9CD467D112B7F21F4"
710 DATA "7E010500EDB0C11095C9"
719 REM BLOCK 7
720 DATA 31880,112,14442
730 DATA "2AF27E3AF17EFE00C847"
740 DATA "C5E55E2356EBE5CD0E7E"
750 DATA "FE3D200901380011587F"
760 DATA "CD2C7EE12CE5D1E1E5D5"
770 DATA "732372EBCD0E7EFE3920"
780 DATA "06E1E1E1E147C9FE3820"
790 DATA "09013D0011687FCD2C7E"
800 DATA "E1E53E16BD201BC1E1E5"
810 DATA "C501907EE5D1AFED42E5"
820 DATA "C1D5E1132BEDB821F17E"
830 DATA "35233434E1E12323C110"
840 DATA "9BC9"
849 REM BLOCK 8
850 DATA 32070,122,12907
860 DATA "3AF67EFE02283A2AF47E"
870 DATA "FE002003251801 24CD0E"
880 DATA "7EFE3828073E0232F67E"
890 DATA "1821E52AF47E01380011"
900 DATA "587FCD2C7EE122F47E01"
910 DATA "3A0011607FCD2C7E3AF6"
920 DATA "7E32F77EC92AF47E2CCD"
930 DATA "0E7EFE38280A2AF77E2F"
940 DATA "E60132F67EC9E52AF47E"
950 DATA "013800115 87FCD2C7EE1"
960 DATA "22F47E013A0011607FCD"
970 DATA "2C7E3AF77E2FE60132F6"
980 DATA "7EC9"
989 REM BLOCK 9
990 DATA 32200,52,6911
1000 DATA "210000FE002806191100"
1010 DATA "001804ED525059E5EBDD"
1020 DATA "E5D1D5E5C5CDB503C1E1"
1030 DATA "DDE1D1 19D5E56069D1D5"
1040 DATA "AF1BED523803E118E2E1"
1050 DATA "E1C9"
1059 REM BLOCK 10
1060 DATA 32270,23,1989
1070 DATA "C5E526002929292 9C1"
1080 DATA "C54806000 9010058097E"
1090 DATA "E1C1C9"
1099 REM BLOCK 11
1100 DATA 32300,86,7782
1110 DATA "C5D5E526002929292929"
1120 DATA "C1C54806000901005809"
1130 DATA "DDE1D1C1D5DDE571E1E5"
1140 DATA "7D21004001000857CB3F"
1150 DATA "CB3FCB3FFE01200109FE"
1160 DATA "022002090987878 7477A"
1170 DATA "90EB6F26002929292929"
1180 DATA "19C1D14806000906081A"
1190 DATA "77132410FAC9" ■
```



Neues aus Vaterstetten:

Mit dem iwt-Programm auf die Zukunft programmiert!

Der C 64 bietet vielseitige grafische Möglichkeiten. Dieses Buch gibt Informationen wie man Grafikfunktionen anwendet – Informationen, die man im Commodore-Handbuch nicht findet. Ausgehend von Grafiken mit den›festen‹Grafik-Zeichen wird systematisch zu den anspruchsvolleren Möglichkeiten, illustriert durch typische Beispiele, geführt.

138 S. Spiralh. DM 38,–

Eine Hilfestellung für wirtschaftliche Entscheidungen sind Programmsammlungen, die die guten Grafik- und Farbmöglichkeiten des Computers nutzen. Diagramme, Sprites, optische Darstellungen von Simulationen werden eingesetzt, die die Ergebnisse verdeutlichen. Die finanzmathematischen Grundlagen sind zu jedem Programm beschrieben.

224 S. Spiralh. DM 38,–

Dieses Buch bietet eine systematische Einführung in die Programmiersprache BASIC. Außer vielen kleineren Programmen zur Illustrierung der BASIC-Anweisungen gibt es eine umfangreiche Programmsammlung zu den verschiedensten Themenbereichen. Die besonderen Fähigkeiten des C 64 werden mit vielen Programmbeispielen erläutert.

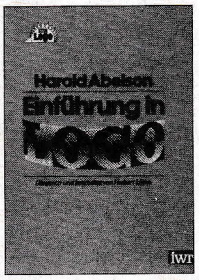
356 S. Spiralh. DM 56,–

Wer hat nicht bereits verzweifelt versucht, das ›Computerchinesenglisch‹ zu verstehen? Hier hilft das Wörterbuch der Computerei mit seinen über tausend Begriffen. Außerdem sind die wichtigsten Begriffe erklärt. Ein handliches Nachschlagewerk für jeden, der sich mit Computerei beschäftigt.

144 Seiten. Kart. DM 32,–

Die Programme sind in TI-BASIC geschrieben, verwenden die Grundkonfiguration des Computers und machen ausgiebig Gebrauch von den Farbgrafikmöglichkeiten sowie dem Tongenerator. Sie sind so angelegt, daß der Einsteiger schnell Erfolge erzielt, der Fortgeschrittene aber die Spielprogramme nach Belieben ergänzen, erweitern oder variieren kann.

190 Seiten. Kart. DM 38,–



LOGO besitzt wichtige Eigenschaften moderner Programmiersprachen. Wesentlich bei LOGO ist die ›Igel-Grafik‹. Mit einfachen Befehlen und Programmen können komplexe Zeichnungen erstellt werden. LOGO ist eine interpretierende Sprache, so können alle Funktionen und Programme ohne Wartezeit ausgeführt werden.

186 S. Spiralh. DM 42,–

Ich bin neugierig auf Ihr Gesamtprogramm! Senden Sie mir umgehend

☐ Ihren neuesten Computer- und Elektronik-Literaturkatalog.
☐ Erbitte Unterlagen über Ihr umfangreiches Software-Programm.
☐ Ich interessiere mich für Ihre ROBOTIK-Idee.
☐ Ich möchte mit D.A.T.A.BOOKS Zeit und Geld sparen.

Name/Vorname

Firma/Abt.

Tel. Beruf

Straße/Hausnr.

PLZ/Ort

IWT-Verlag, Vaterstetten
Der Fachverlag für Information, Wissenschaft, Technologie
Dahlienstraße 4, 8011 Vaterstetten, Post Baldham, Tel. (0 81 06) 3 10 17
Ausl. Schweiz: Thali AG, Buchhandlung u. Verlag, CH-6285 Hitzkirch, Tel. 041/85 28 28

Drei für einen — einer für drei

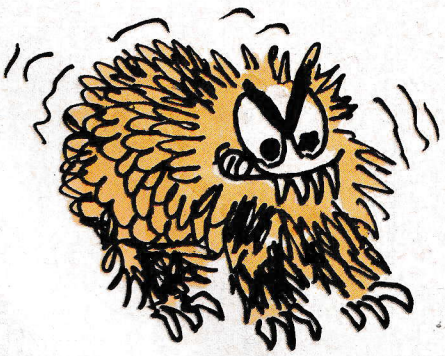
Minis für 1 K ZX81

Viele ZX-Besitzer staunen nicht schlecht, wenn sei nach dem Kauf ihres Gerätes feststellen, welch enge Programmiergrenzen ihnen mit 1 K freiem RAM gesetzt sind.

Dem User tun sich an dieser Stelle zwei Wege auf: Zum einen kann er sich in den berühmten Schmollwinkel zurückziehen und warten, bis er das nötige Kleingeld, das zum Kauf einer Speichererweiterung gebraucht wird, zusammengetragen hat. Zum anderen kann er sich hinsetzen und versuchen, das beste aus 1 K Speicherplatz zu machen. 3 Programme von Usern, die nicht resigniert haben, möchten wir in dieser Ausgabe von COMPUTER Praxis vorstellen:

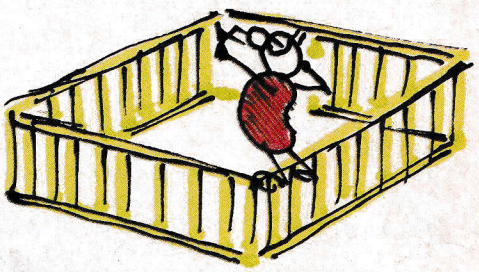
Das erste der drei 1 K Spiele nennt sich Android Attack und kommt von Michael Kincaid und David Chapman. Es geht darum, so viele Androiden wie möglich abzuschießen, bevor drei von ihnen entkommen. Man bewegt seine Kanone mit den Tasten U für hoch und N für runter. Gefeuert wird mit der Taste F wie 'Fracksausen'.

Iain McWilliam's Mole ist ein Reaktionsspiel. Der Held dieses Spieles ist ein Maulwurf, der tief in der Erde seinen Weg gräbt. Unglücklicherweise ist er blind und muß deshalb mit den Tasten N und M geleitet werden. Für jeden Schritt der unseren Maulwurf ein Stück weiterbringt, ohne ihn gegen die Wand rennen zu lassen, gibt es einen Punkt.



Catch-a-Bag-of Swag von Julian Downes gibt uns die Möglichkeit, endlich mal Geld, das in Mengen von der Bank fällt, einfach aufzufangen. Den Eimer, der dazu notwendig ist, steuert man mit den Tasten "5" und "8". Die Geldsäcke haben verschiedene Werte, die durch Zufallsgenerator ermittelt werden. Versucht so viele Geldsäcke wie möglich zu 'catchen', um eine angemessene Rendite zu erreichen. Nach 15 Geldsäcken startet das Spiel von neuem.

Wir wünschen Euch viel Spaß beim Spielchen mit den 'Minis' für euren Mini, den ZX81.



Android Attack

```
 1 LET K = 0
 2 LET C = 0
 3 LET D = 1
 5 LET P = 4
 6 LET S = 4
20 FOR T = D TO 28
30 PRINT AT S,T;" {INV. ON}  {INV. OFF}{GRAPHICS}6"
35 PRINT AT P,25;"<"
45 LET V = 0
50 IF INKEY $ = "U" THEN LET V = -4
60 IF INKEY $ = "N" THEN LET V = 4
65 IF INKEY $ = "F" AND P = S THEN GOTO 200
70 IF P+V < = 3 THEN LET V = 0
 85 PRINT AT P,25;" "
 90 LET P = P+V
100 NEXT T
105 LET C = C+1
106 IF C = 3 THEN GOTO 230
110 LET S = INT (RND*4)*4+4
120 IF S > = 16 THEN LET S=4
130 GOTO 20
200 PRINT AT S,T;"{4SPACE}"
210 LET K = K+1
215 IF K> 10 THEN LET D = D+2
220 GOTO 110
230 PRINT "SCORE = "; K
```

Mole

```
2 FOR A = 1 TO 10
3 PRINT "{INV ON} {INV OFF}{5SPACE}
  {INV ON} {INV OFF}"
4 NEXT A
5 LET P = 3
6 LET S = 0
7 LET A = INT (RND*5+1)
8 GO SUB 20
9 GO SUB 24
10 IF R = 128 THEN GO TO 30
11 IF RND<.7 THEN GO TO 8
12 FOR A = 1 TO 4
13 SCROLL
14 PRINT AT 9,0;"{INV ON} {INV OFF}{5SPACE}{INV ON}
   {INV OFF}"
15 IF RND<.3 AND A>1 AND A<4 THEN PRINT AT 9,P;
   "{INV ON} {INV OFF}"
16 GO SUB 24
```

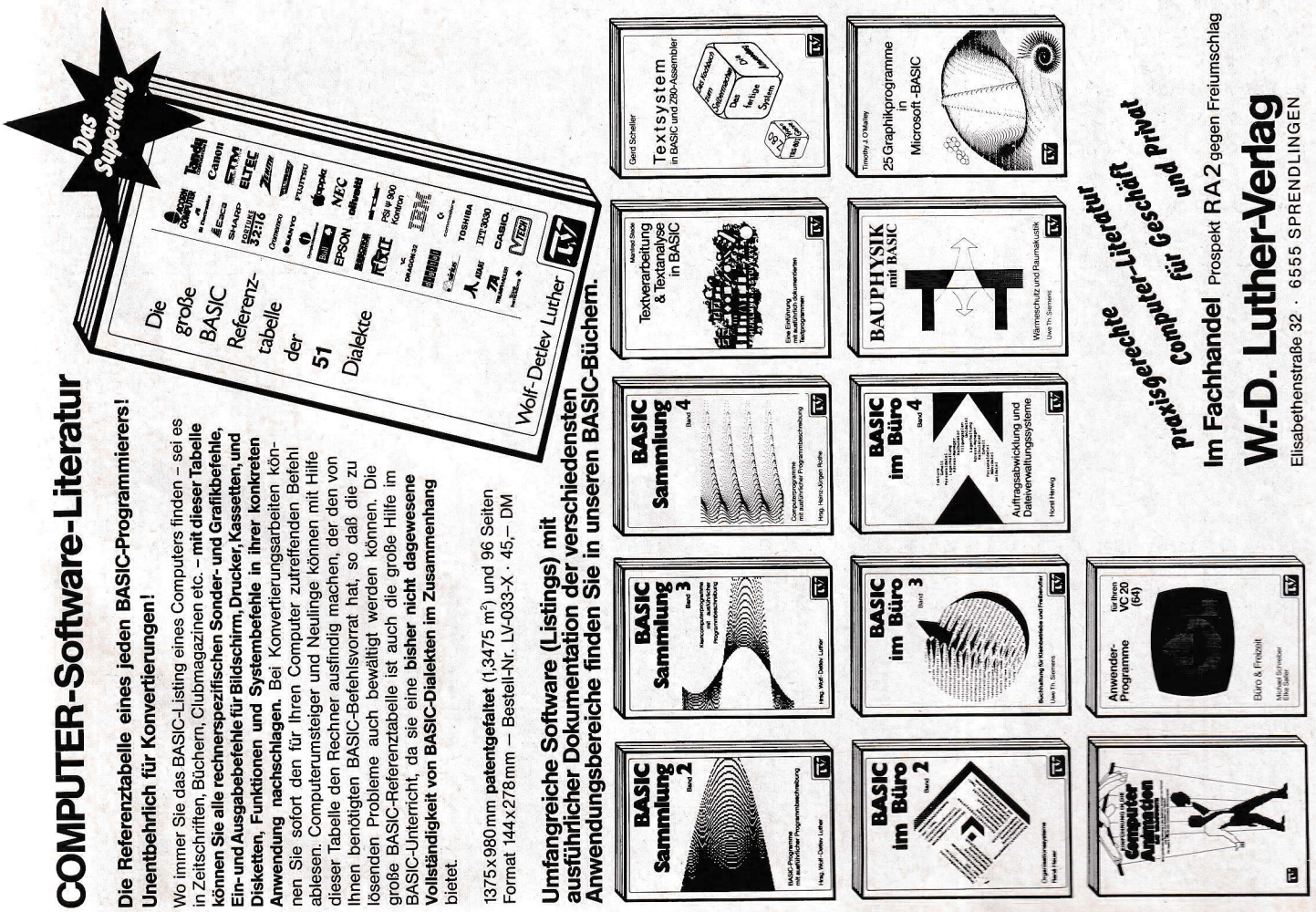
```
17 IF R = 128 THEN GO TO 30
18 NEXT A
19 GO TO 7
20 SCROLL
21 PRINT AT 9,0;"{INV ON}{7SPACE}{INV OFF}"
22 PRINT AT 9,A;" "
23 RETURN
```

```
24 PRINT AT 3,P;
25 LET R = PEEK (PEEK 16398 + PEEK 16399)
26 PRINT"+"
27 LET S = S+1
28 LET P = P +(INKEY $ ="M") - (INKEY $ ="N")
29 RETURN
30 PRINT AT 4,P; S
```

Swag

```
10 LET H = 0
20 LET X = 0
30 LET S = 0
40 FOR M = 1 TO 15
50 LET A = INT (RND * 12)+2
60 LET B = (INT (RND * 5)) * 10
70 PRINT AT 1,1; " {GRAPHICS ON} 8 {GRAPHICS OFF}
   {INV.ON}£££££££££££{INV.OFF} {GRAPHICS ON} 5 "
80 FOR Y = 1 TO 9
90 PRINT AT Y,A; "{INV.ON}£{INV.OFF}"
100 IF INKEY $ = "5" AND X<>0 THEN LET X = X-1
110 IF INKEY $ = "8" AND X<>11 THEN LET X = X+1
120 PRINT AT 10,X; "  {GRAPHICS ON} 8 {GRAPHICS OFF}
    {INV.ON} {INV.OFF} {GRAPHICS ON} 5  "
```

```
130 PRINT AT Y,A; " "
140 NEXT Y
150 PRINT AT 10,A; "INV.ON}£{INV.OFF}"
160 IF A <> X+2 THEN GOTO 200
170 LET S = S+B
180 PRINT AT 0,2; "SWAG = £";S
190 GOTO 210
200 PRINT AT 10,A;"  "
210 NEXT M
220 CLS
230 IF H<S THEN LET H = S
240 PRINT AT 0,12;"HI-SWAG = £";H
250 GOTO 30 ■ G.K.
```

SOFTWARE IM TEST

Jetpac

Der Spectrum 16 K-Besitzer muß auf seinem Heimatplaneten eine Rakete zusammenbauen, die er mit Treibstoff füllt, um zu seiner Reise in die Unendlichkeit des Raumes aufzubrechen. Beim Sammeln der Treibstoffkanister wird er jedoch pausenlos von außerirdischen Wesen attackiert, denen er entweder einen Schuß aus der Photonenkanone verpaßt oder ausweicht. Zusätzlich hat unser Held noch die Möglichkeit, herumliegende Schätze einzusammeln; das bringt Bonuspunkte. Mit jeder erreichten Spielstufe werden die Aliens gewitzter und das Bemühen um ein Auftanken der Rakete aussichtsloser.

Die Graphik dieses Spieles ist erfreulich gut gelungen, und nur der Sound kommt bei diesem Spiel nicht über das Mittelmaß hinaus, da er sich, wie bei den meisten Spielen dieser Gattung darauf beschränkt, Knall- Zisch- und Bumgeräusche zu produzieren.

Kurz und bündig: Jetpac ist ein Spacegame, das sich im Gegensatz zu den meisten seiner Genossen durch ein gehöriges Maß an Spielwitz auszeichnet und das gerade für den Actionfreund unter den Spectrum-Usern so manche vergnügte Stunde garantiert.

Graphik:

Sound:

Spielidee:

Testergebnis:

Gobbleman

Selbst der sprichwörtliche Laie in Sachen Computerspiele wird nach dem Laden dieses Spieles in seinen 16 K-Spectrum sofort sagen: »Das kennst Du doch!?«. Nicht weiter verwunderlich, denn was sich da auf dem Bildschirm unter dem Namen Gobbleman präsentiert, ist nichts anderes als eine weitere Version des hinlänglich bekannten PACMAN.

Leider ist die Graphik wegen der hohen Auflösung so klein geraten, daß man während des Spiels schon eine Lupe zur Hand nehmen muß, um alle Einzelheiten zu erkennen. Trotzdem kann das Spiel wegen seines Schwierigkeitsgrades, der es im übrigen fast unmöglich macht, Gobbleman über die Tastatur zu steuern, schon eine Herausforderung an den Ehrgeiz eingefleischter Spieler sein.

Graphik:

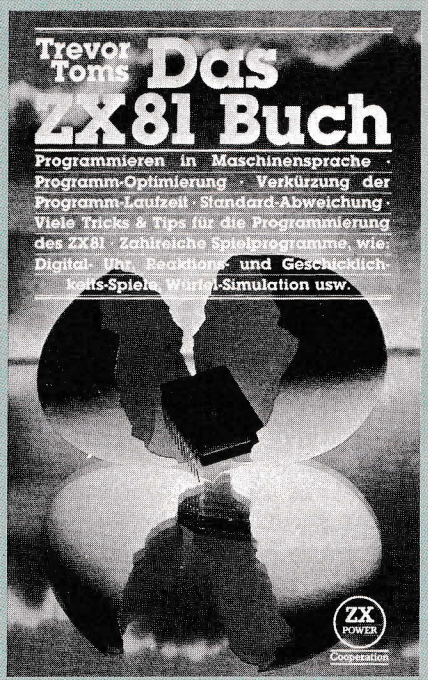
Sound:

Spielidee:

Testergebnis:

Für Einsteiger

Wer verstehen will, wie sein ZX 81 Programme und Programmanweisungen verarbeitet, wird in diesem Buch mit für Anfänger nützlichen Tips und Tricks für BASIC versorgt.



Das ZX-81 Buch
Toms
Verlag Hueber Software
29,80 DM
123 Seiten
ISBN 3-19-008203-0

Neben Grundlagen, aus denen ihr euch eigene Spiel und Spielvarianten aufbauen könnt, finden sich hier Spiele, an denen der Autor demonstriert, wie sich bestimmte Programmiermethoden auf Laufzeitverhalten und Speicherplatzbedarf auswirken.

Spielprogramme, von denen viele auch für die 1K-Version geeignet sind, und Informationen finden sich gut gemischt.

Die Maschinenprogrammierung allerdings, auf die auch auf dem Einband hingewiesen wird, kommt zu kurz: Gezeigt wird lediglich, wie ein Maschinenprogramm im Speicher angesprochen werden kann.

Anstelle des auf 27 Seiten ausgewalzten Adventure-Spiels wäre uns und Euch z.B. viel mehr mit der Erklärung einiger Systemadressen gedient. Auch ein Schlagwortregister wäre (wie bei jedem Sachbuch) mehr als angebracht.

Für den Einsteiger jedoch ist dieses Buch interessant und nützlich.

Thok

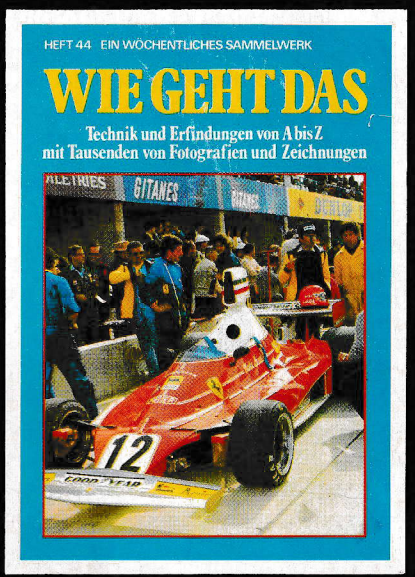
WIE GEHT DAS

Das neue Sammelwerk, in dem Sie alles Wissenswerte aus Naturwissenschaft und Technik finden. WIE GEHT DAS beschreibt Maschinen, Technologien, Erfindungen. Die ganze Welt der Technik, alphabetisch geordnet. Zum Nachschlagen und Studieren. So wird WIE GEHT DAS Ihr einmaliges Nachschlagewerk. Wöchentlich. Beim Zeitschriftenhändler.

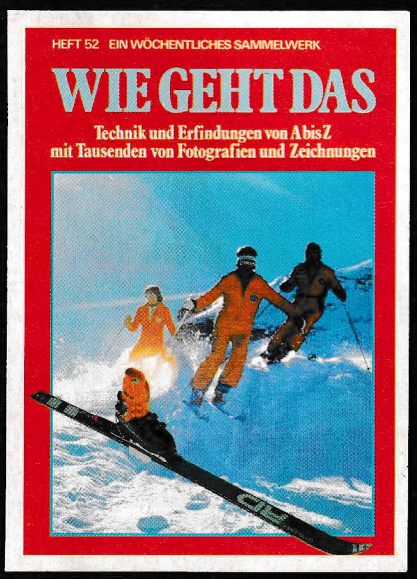
von Elektronik in der Medizin bis Fahrgestell

von Flugzeugpropeller bis Freilauf

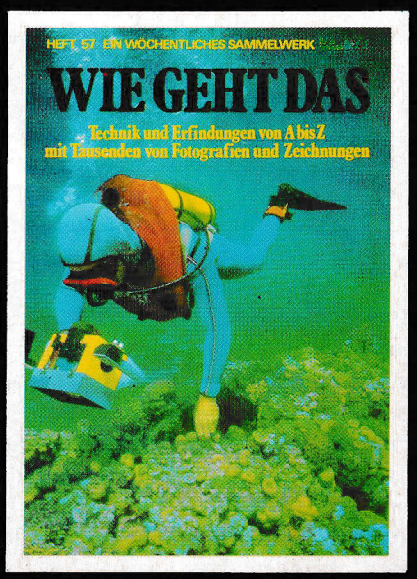
von Metalldetektoren bis Mikroskop



von Raumsonden bis Rennwagen



von Sirene bis Spektroskopie



von Taucheranzug bis Teilchenphysik

von Trickeffekte bis Überschallflug

Woher nehmen und nicht stehlen?

Neue Peripherie ist noch immer viel zu wenig in Sicht. Machen wir uns darum einen normalen Cassetten-Recorder für unseren TI einsetzbar!

Wer sich einen Computer zulegt, wird früher oder später den Wunsch verspüren, Software nicht nur für die Zeit parat zu haben, in der der Rechner eingeschaltet ist. Oder der neue User hat, durch hilfreiche Unterstützung der Werbung, festgestellt, daß es Software fertig zu kaufen gibt. Und zwar entweder auf Disketten oder auf Cassetten.

Damit sind wir beim Thema. Da Texas Instruments so freundlich war, seine Lager zu räumen, bevor sich die Firma aus der Home-Computer-Branche zurückzog, sitzen heute viele User ohne Massen-Speicher an ihrem Rechner. Disketten-Laufwerke und die dazugehörigen Controller sind für die Expansion-Box nicht mehr zu bekommen — wenn man überhaupt noch das Glück hatte, eine Box zu ergattern.

Entsprechender Ersatz von anderen Herstellern, der mit unseren TI 99/4A kompatibel ist, ist auf dem Markt nicht erhältlich. Also greife man auf den Cassetten-Recorder zurück.

Sofort ergibt sich das nächste Problem: Anschluß-Kabel Rechner-Recorder sind nicht mehr oder nur zu horrenden Preisen zu bekommen. Ein Grund für uns, euch eine Schaltskizze anzubieten. Anhand der hier abgedruckten Skizze könnt ihr euch dieses Kabel selber bauen. Alles was ihr braucht, bekommt ihr im Fachhandel:

▷3 Klinken-Stecker 3,5 mm, 2-polig
▷2 Klinken-Stecker 2,5 mm, 2-polig
▷1 V9-Stecker (female)
▷1,5 m 6-phasiges Kabel
und den obligatorischen Lötkolben.

Das Kabel reicht, um die Möglichkeiten des TI 99/4A voll zu nutzen, er ist nämlich in der Lage mit zwei Recordern zu arbeiten! Die Pins des V9-Steckers sind wie folgt belegt:

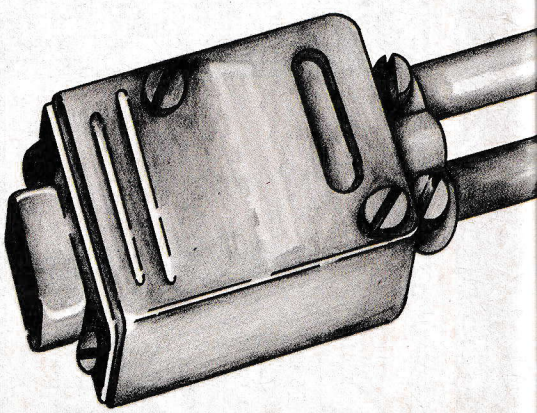
Pin 1 und 2: Fernsteuerung Recorder 1 (CS1)
Pin 3 und 5: Aufnahme Recorder 1 und Recorder 2 (CS1 und CS2)
Pin 6 und 7: Fernsteuerung Recorder 2 (CS2)
Pin 8 und 9: Abspielen Recorder 1 (CS1)
Pin 4 ist nicht belegt. Wenn ihr nur einen Recorder anschließen wollt, braucht ihr die Pins 6 und 7 auch nicht zu belegen. Ihr kommt dann mit einer einzigen Leitung aus.

Beachten müßt ihr noch, daß die Pins 3 und 9 jeweils die Abschirmung für Aufnahme bzw. Abspielen sind. Wie die Fernsteuerung gepolt werden muß, hängt vom anzuschließenden Recorder ab. Hier hilft nur das Experiment.

Welcher Recorder?

Eine nicht selten gehörte Frage ist, welcher Recorder denn nun der beste ist. Unsere Erfahrung: Ein einfacher Mono-Recorder reicht völlig. Solche Geräte sind ab ca. 60,— DM im Handel. Alle Versuche mit teuren Stereo-Geräten oder sogar Spulen-Tonbandgeräten verliefen negativ.

Der zweite Punkt, der beachtet werden muß: nur gutes Bandmaterial benutzen. Billig-Cassetten oder alte Musik-Cassetten haben nur ein Ergebnis, Frust. Diese alten Bänder verlieren nämlich mit der Zeit ihre Magnet-Schicht und damit die darin enthaltene Information. Bei Musik-Aufnahmen fällt das unter Umständen noch nicht einmal auf. Aber der Computer mault sofort "ERROR IN DATA DETECTED", Fehler in den Daten gefunden. Das bedeutet, die gespeicherten Daten sind zum Teufel. Was dem User bleibt, ist mal wieder Frust.

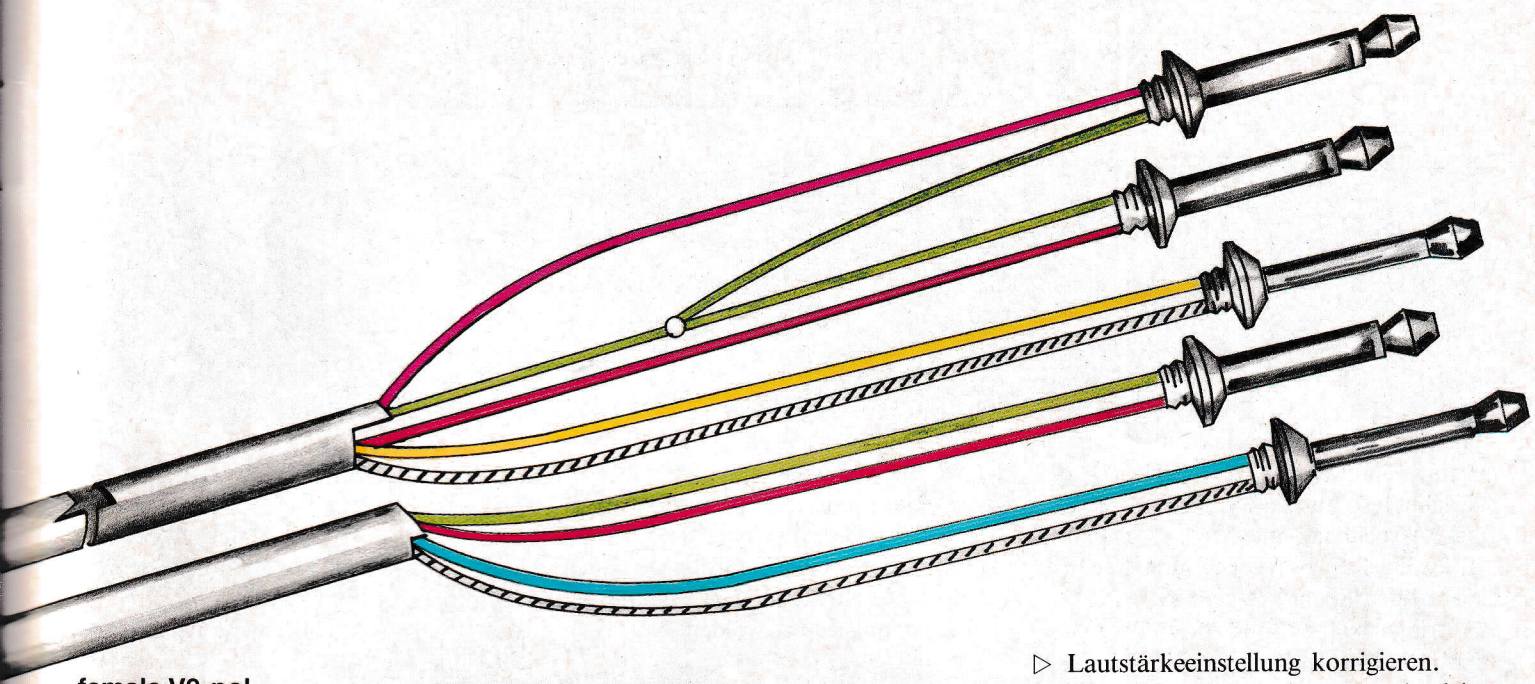


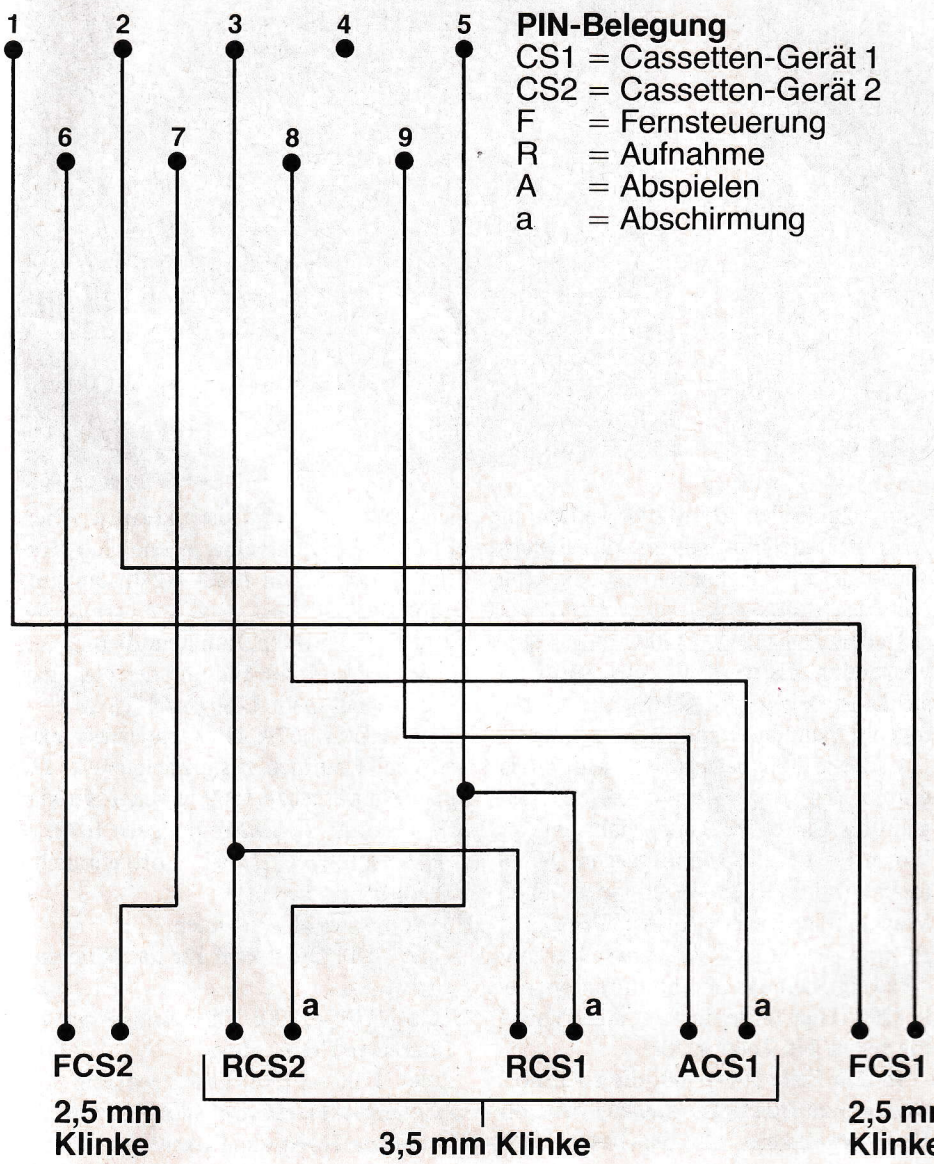
So sieht unser Anschlußkabel in der Version für zwei Cassettenrecorder aus.

Aus diesem Grund sollte man auch von allen wichtigen Programmen eine "BackUp"-Kopie, das heißt eine Sicherungs-Kopie, anfertigen. Und zwar auf einer anderen und nicht auf der gleichen Cassette. Wenn euch jetzt ein Programm durch Altersschwäche der Cassette ins Nirwana abhaut, nehmt ihr locker eure BackUp aus dem Regal und könnt weiterarbeiten (oder spielen).

Cassetten sollte man übrigens nicht auf Magneten oder in der prallen Sonne lagern. Hitze und Magnetfelder haben die unangenehme Eigenheit, Tonband-Material zu löschen und Wärme verformt außerdem das Gehäuse einer Cassette unter Umständen bis hin zur Unbrauchbarkeit.

Und dann gibt es Tage, da will sich ein Programm, das man schon so oft erfolgreich geladen hat, nicht in den Rechner holen lassen. Auch die BackUp hilft nicht. Auch hierfür haben wir einige Tips für euch. Helfen auch die nicht, weiß keiner mehr weiter.





▷ Lautstärkeeinstellung korrigieren.
▷ Anschlüße überprüfen. Manche leiern aus.
▷ steht der Recorder zu dicht am Fernseher (Monitor)? Die Ausstrahlungen der Bildröhre können den Ladevorgang beeinträchtigen.
▷ Tonköpfe reinigen. Dabei sollte man lieber Isopropanol, ein Alkohol, den ihr in der Apotheke bekommt und Watte-Stäbchen benutzen als teure Reinigungs-Bänder, die euch unter Umständen den Tonkopf zusätzlich mit Schmutzpartikeln verkleben.
▷ Als letzte Maßnahme: Vorsichtige Korrektur der Spurlage

Im Normalfall hat der TI 99/4A-User aber keine Probleme, Programme auf Cassetten zu speichern und sie von dort wieder einzuladen. Wenn ihr darauf achtet, daß euer Bandmaterial in Ordnung ist, werdet ihr so manches interessante Daten-Verwaltungs-Programm schreiben können. Denn mit dem TI kann man nicht nur Programme sondern auch Daten auf Cassette abspeichern. Natürlich gehört auch etwas Geduld dazu, denn die Baud-Rate, das ist die Zahl der übertragenen Bits pro Sekunde, ist bei dem TI 99 leider ziemlich niedrig.

Wer aber diese Geduld nicht aufbringen kann oder will, der muß auf andere Weise Geduld beweisen. Denn die angekündigten Floppy-Controller sind leider immer noch nicht auf dem Markt. Und dabei wird es wohl auch noch eine ganze Weile bleiben. ■

H.U.

EIMERKETTE

Der Zauberlehrling in XBASIC

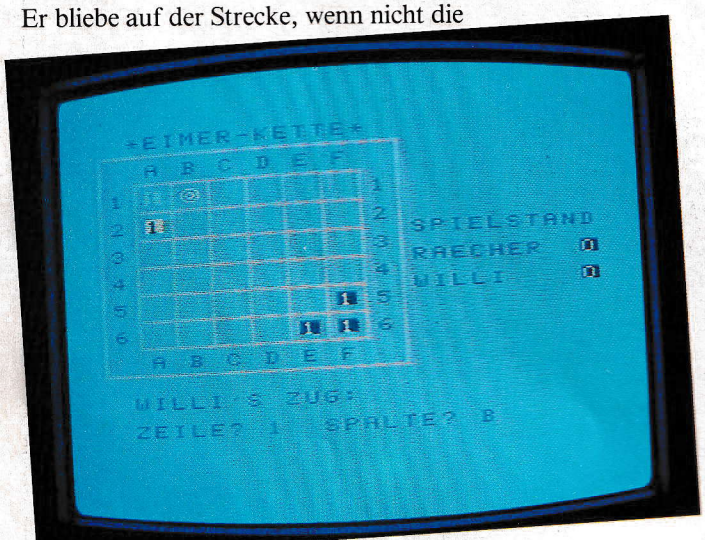
Bei diesem Listing handelt es sich um eine Art Brettspiel. Jedes Feld des Brettes bedeutet einen Eimer, und es ist Ziel des Spiels, die Eimer unter Kontrolle zu bekommen. Es gilt die einfache Regel: "Ein Eimer ist mein, wenn was von mir drinliegt". Man braucht also nur Zug für Zug an die Kübel zu spenden, und der TI zeigt brav an, wieviel im Eimer liegt und welchem der beiden Spieler es gehört.

Prima, nur — wo bleibt da der Witz? Er bliebe auf der Strecke, wenn nicht die Eimer die magische Eigenschaft hätten, überzulaufen, sobald sie den Rand voll haben. Man kann also seinen Obolus nach und nach an freie Eimer entrichten, oder man kann einen, den man schon in seinen Besitz gebracht hat, durch steten Tropfen zum Überquellen bringen. Und das geht rasch. In der Ecke stehende Behälter gehen schon hoch, wenn man sie ein zweites Mal bedenkt, Eimer am Rand werden bei der dritten Füllung zu Springbrunnen und die Mittelfeldkübel reagieren auf die Anzahl 4 allergisch. Dann nämlich speien sie ihren Inhalt in die Nachbareimer, und bringen diese samt gegnerischem Inhalt in den eigenen Besitz. Mit etwas Geschick erreicht man scheinbar endlose Ketten-Reaktionen.

Das Spiel kann zu zweit oder allein gegen den TI 99 gespielt werden, oder ihr laßt den Rechner einfach gegen sich selbst antreten.

Zum Listing

Ihr braucht einen TI 99/4A und Extended BASIC. Das Spiel ist mit 8181 Bytes relativ kurz, wenn man bedenkt, was es leistet. Achtet bitte darauf, daß einige Zeilen sowohl Groß- als auch Klein-Buchstaben enthalten. Sie müssen genau so eingegeben werden!



In den Zeilen 100—200 werden die Arrays (Felder) initialisiert, das Bezugs-Array belegt, die Farben und die Character gesetzt.

In den Zeilen 210—350 wird das Spiel vorbereitet. Es werden, wenn nötig, Instruktionen gegeben, die Spieler-Anzahl und ihr Namen erfragt.

In Zeile 360 beginnt das Hauptprogramm. Es ist zwei-geteilt. Der erste Teil geht bis Zeile 580. Hier habt ihr das 'normale Spiel' Mann gegen Mann (oder Frau gegen Frau oder so...). Wenn ihr gegen den TI spielt oder den Rechner gegen sich selbst antreten laßt, verzweigt sich das Programm in die Zeile 1290. Hier findet ihr die 'Automatik'. Sie reicht bis Zeile 1570.

In den Zeile 590—660, die auch noch zum Hauptprogramm zählen, wird das Ende des Spieles mit der Feststellung, wer gewonnen hat, abgehandelt. Das Programm bleibt hier stehen, bis eine Taste gedrückt wird. Das Spiel wird mit einem STOP abgebrochen, wenn ihr 'N' drückt. Sonst kommt sogleich die Revanche-Runde.

Es folgen die Sub-Routinen

Zeile 670— 780: Das Spielbrett wird aufgebaut.

Zeile 790— 850: Regelt die Wasserzuteilung an die Eimer.

Zeile 860—1030: Hier wird geprüft, ob ein Eimer überläuft.

Zeile 1040—1240: Instruktionen. Diese Sub-Routine hat eine eigene Sub-Routine in den Zeilen 1250—1270, um 'umzublättern'.

Zeile 1580—1600: Punkteaddierer.

Zeile 1610—1640: Füllung der Nachbar-Eimer bei einem Überlauf.

Zeile 1650—1670: Hier sind zwei 'akustische' Routinen zusammengefaßt. Die erste, in der Zeile 1660, erzeugt die kleine Melodie, die zweite, in Zeile 1670, ist eine einfache aber oftgebrauchte Tonfolge.

Zum Schluß die Daten

Zeile 1690: Die Daten für das Referenz-Array.

Zeile 1700—1720: Die Strings für die Character-Definierung.

Zeile 1730—1790: Die Frequenz und Tonlängen für die Musik.

Und nun: Hoch die Tassen! Äh, Eimer.

```
100 REM INITIALISIERUNG
110 CALL CLEAR :: CALL SCREEN(5)::
    DISPLAY AT(12,9):"EIMERKETTE" :: RANDOMIZE
120 GOSUB 1660
130 DIM W(6,6),B(6,6),O(6,6),P(6,6),CN$(2),V(2),D(2)
140 FOR N=1 TO 6 :: READ W(N,1):: W(N,6)=W(N,1)::
    FOR J=2 TO 5 :: W(N,J)=W(N,1)+1
150 NEXT J :: NEXT N :: CALL COLOR(14,10,1)
160 DEF RN(X)=INT(RND*X+1)
170 CALL COLOR(3,12,2):: CALL COLOR(9,14,1)::
    CALL COLOR(10,14,1):: CALL COLOR(13,2,12)
180 FOR BU=128 TO 135 :: CALL CHARPAT(BU-80,A$)::
    CALL CHAR(BU,A$):: CALL CHAR(BU-16,A$):: NEXT BU
190 FOR N=97 TO 107 :: READ N$ :: CALL CHAR(N,N$)::
    NEXT N
200 CALL CHAR(136,"0000001818")::
    CALL CHAR(137,"00003C425A5A423C")::
    CALL CHAR(138,"3C4299A5A599423C")
210 REM SPIELVORBEREITUNG
220 DISPLAY AT(10,4):"WILLKOMMEN ZUM SPIEL"
230 FOR BU=1 TO 3 :: CALL SOUND(200,220,4)::
    CALL SOUND(400,440,2):: NEXT BU
240 DISPLAY AT(22,1):"BRAUCHST DU INSTRUKTIONEN?" ::
    CALL KEY(3,KEY,ST):: IF ST=0 THEN 240 ELSE
    IF KEY=74 THEN GOSUB 1050
250 CALL CLEAR :: FOR BU=1 TO 50 ::
    CALL SOUND(-50,880+10*BU,3):: NEXT BU
260 DISPLAY AT(2,2):"SPIELERANZAHL [p,q,r]?" ::
    CALL KEY(3,KEY,ST):: IF ST=0 OR KEY<48 OR KEY>50
    THEN 260 ELSE G=KEY-48
270 CALL SOUND(1000,-2,1)
280 IF G<2 THEN CN$(2)=" TI 99" ::
    IF G<1 THEN CN$(1)=" 99pp" :: GOTO 340
290 FOR N=1 TO G
300 DISPLAY AT(4*G,2):"NAME VON SPIELER ";
    CHR$(112+N);" ?" ::
    ACCEPT AT(4*G+2,4)VALIDATE(UALPHA):CN$(N):: L=0
310 CN$(N)=SEG$(CN$(N),1,8)
320 NEXT N
330 CALL SOUND(600,440,2):: CALL SOUND(300,220,5)
340 CALL CLEAR :: GOSUB 680 :: IF RND<.5 THEN K=1 ::
    GOTO 370
350 K=2 :: I=-1
360 REM HAUPTPROGRAMM
370 CALL HCHAR(20,1,32,96)
380 L=L+1 :: S=1-(K=1)
390 IF K>10 THEN K=10 ELSE IF K<1 THEN K=1
400 A=H(K):: V3=(L>2)
410 DISPLAY AT(20,4):CN$(K);"'S ZUG:" ::
    IF K=2 AND G<2 THEN 1290
420 IF K=1 AND G<1 THEN 1290
430 DISPLAY AT(22,4):"ZEILE?"
440 CALL KEY(3,KEY,ST)::
    IF ST=0 OR KEY<49 OR KEY>54 THEN 440
450 Y=KEY-48 :: N=KEY+64 ::
    DISPLAY AT(22,11):CHR$(N)
460 DISPLAY AT(22,14):"SPALTE?"
470 CALL KEY(3,KEY,ST)::
    IF ST=0 OR KEY<65 OR KEY>70 THEN 470
480 X=KEY-64 :: DISPLAY AT(22,22):CHR$(KEY)::
    IF X=-44 THEN 530
490 GOSUB 1670
500 IF O(Y,X)=K OR B(Y,X)=0 THEN 540
510 DISPLAY AT(24,1)BEEP:"VERBOTENE POSITION
     - NOCHMAL"
```

TI 99/4A ***Erweitern Sie Ihr System*** **TI 99/4A**

Nutzen Sie den Telefonkontakt und fordern Sie Unterlagen an: **040 - 668 22 56**

- *volle Kompatibilität zum TI 99/4A*
- *softwaremässige Anpassung*
- *alle Komponenten einzeln funktionsfähig*
- *Erweiterung durch einfaches Aneinanderstecken*

mit System!

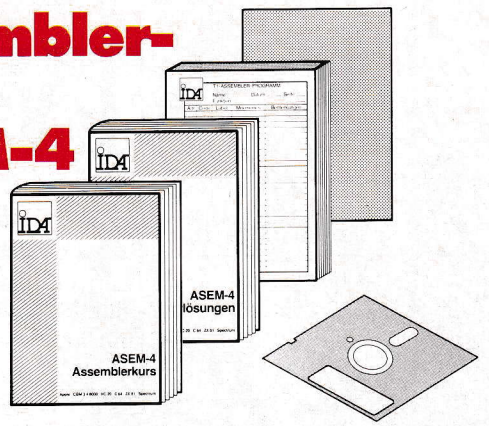
Schnittstellen - Speichererweiterungen - Disketten-Steuerungen - Disk-Laufwerke

atronic Bunke & Vernunft oHG
Postfach 700 124
2000 Hamburg 70

```
520 CALL SOUND(300,440,3):: CALL SOUND(900,220,3,
530 CALL HCHAR(20,1,32,160):: GOTO 410
540 B(Y,X)=B(Y,X)+1 :: O(Y,X)=K :: F=1 :: GOSUB 800 ::
    F=0 :: IF B(Y,X)=1 THEN D(K)=D(K)+1
550 Z=1 :: IF B(Y,X)=W(Y,X)THEN GOSUB 870
560 IF Z<>1 THEN 600
570 IF K=1 THEN K=2 :: GOTO 370
580 K=1 :: GOTO 370
590 REM SPIELENDE
600 CALL HCHAR(20,1,32,96)::
    DISPLAY AT(22,3):CN$(K);" HAT GEWONNEN!"
610 FOR BU=1 TO 3 :: CALL SOUND(100,500,1)::
    CALL SOUND(200,800,1):: NEXT BU
620 V(K)=V(K)+1 :: GOSUB 800 :: I=(K=2):: V3=0
630 FOR BU=1 TO 50 :: CALL SOUND(-50,880+10*BU,4)::
    NEXT BU
640 CALL KEY(3,KEY,ST)::
    IF ST=0 THEN 640 ELSE IF KEY=78 THEN CALL CLEAR ::
    STOP
650 CALL CLEAR :: FOR N=1 TO 6 :: FOR J=1 TO 6 ::
    B(N,J)=0 :: O(N,J)=0 :: NEXT J :: NEXT N ::
    D(1)=0 :: D(2)=0
660 L=0 :: I=NOT I :: GOSUB 680 :: K=1-I :: GOTO 370
670 REM SUB - SPIELBRETT-AUFBAU
680 DISPLAY AT(1,4):"*EIMER-KETTE*" ::
    DISPLAY AT(2,2):"abbbbbbbbbbbbbbbd" ::
    DISPLAY AT(3,2):"e  A B C D E F  e"
690 DISPLAY AT(4,2):"e abcbcbcbcbcbd e"
700 DISPLAY AT(5,2):"eqe e e e e e eqe"
710 FOR V5=1 TO 5 ::
    DISPLAY AT(4+2*V5,2):"e fbgbgbgbgbgbh e"
720 DISPLAY AT(5+2*V5,2):"e";CHR$(V5+113);
    "e e e e e e e";CHR$(V5+113);"e" :: NEXT V5
730 DISPLAY AT(16,2):"e ibjbjbjbjbjbk e"
740 DISPLAY AT(17,2):"e  A B C D E F  e" ::
    DISPLAY AT(18,2):"jbbbbbbbbbbbbbbbk"
750 DISPLAY AT(8,19):"SPIELSTAND" ::
    DISPLAY AT(10,19):CN$(1)::
    DISPLAY AT(12,19):CN$(2)
760 DISPLAY AT(10,28):CHR$(ASC(STR$(V(1)))):
    DISPLAY AT(12,28):CHR$(ASC(STR$(V(2))))
770 GOSUB 1660
780 RETURN
790 REM SUB - STEIN SETZEN
800 FOR N=1 TO 6 :: FOR J=1 TO 6
810 IF B(N,J)=0 THEN CALL HCHAR(2*N+3,2*J+5,32)::
    GOTO 840
820 IF O(N,J)=2 THEN VO=80
830 CALL HCHAR(2*N+3,2*J+5,ASC(STR$(B(N,J)))+VO)::
    VO=0 :: CALL SOUND(-100,-8,1)
840 NEXT J :: NEXT N
850 RETURN
860 REM SUB - UEBERLAUF
870 Y=1 :: X=1
880 IF B(Y,X)>=W(Y,X)THEN 920
890 X=X+1 :: IF X<=6 THEN 880
900 Y=Y+1 :: X=1 :: IF Y<=6 THEN 880
910 RETURN
920 CALL SOUND(-200,220,8,230,8,110,6,-7,10)::
    CALL HCHAR(3+Y*2,2*X+5,136)
930 CALL SOUND(200,240,6,230,8,110,4,-7,6)::
    CALL HCHAR(3+Y*2,2*X+5,137)
940 CALL SOUND(-300,260,4,800,0,300,2,-7,3)::
    CALL HCHAR(3+Y*2,2*X+5,138)
950 IF Y>1 THEN Q=Y-1 :: B(Q,X)=B(Q,X)+1 :: U=X ::
    GOSUB 1620
960 IF X>1 THEN U=X-1 :: B(Y,U)=B(Y,U)+1 :: Q=Y ::
    GOSUB 1620
970 IF Y<6 THEN Q=Y+1 :: B(Q,X)=B(Q,X)+1 :: U=X ::
    GOSUB 1620
980 IF X<6 THEN U=X+1 :: B(Y,U)=B(Y,U)+1 :: Q=Y ::
    GOSUB 1620
990 B(Y,X)=B(Y,X)-W(Y,X):: O(Y,X)=K
1000 IF B(Y,X)=0 THEN O(Y,X)=0 ::
    D(K)=D(K)-1
1010 F=1 :: GOSUB 800 :: F=0
1020 Z=1 ::
    IF (D(1)+D(2)>1)AND(D(1)=0 OR D(2)=0)THEN Z=0 ::
    RETURN
1030 GOTO 870
1040 REM SUB - INSTRUKTIONEN
1050 IO=1 :: CALL CLEAR
1060 DISPLAY AT(2,5):"eSPIELANLEITUNGe" ::
    DISPLAY AT(1,5):"abbbbbbbbbbbbbbd"
1070 DISPLAY AT(3,5):"ibbbbbbbbbbbbbbk"
1080 DISPLAY AT(5,1):"BEI DIESEM SPIEL WERDEN AUF" ::
    DISPLAY AT(7,1):"DEM SPIEL BRETT ABWECHSELND"
1090 DISPLAY AT(9,1):"STEINE GESETZT. DABEI DARF" ::
    DISPLAY AT(11,1):"MAN NUR LEERE ODER EIGENE"
1100 DISPLAY AT(13,1):"FELDER BELEGEN. WIRD DIE" ::
    DISPLAY AT(15,1):"MAXIMALE BELEGZAHL UEBER-"
1110 DISPLAY AT(17,1):"SCHRITTEN, LAEUFT DAS FELD" ::
    DISPLAY AT(19,1):"UEBER UND ERGIESST SEINEN"
1120 DISPLAY AT(21,1):"INHALT AUF DIE NACHBARFELDER"
1130 GOSUB 1250 :: CN$(1)="SPp" :: CN$(2)="SPq"
1140 FOR N=1 TO 6 :: FOR J=1 TO 6 :: O(N,J)=0 ::
    B(N,J)=0 :: NEXT J :: NEXT N :: O(2,1)=1 ::
    O(4,3)=2
1150 B(2,1)=2 :: B(4,3)=3 :: GOSUB 680 :: GOSUB 800 ::
    DISPLAY AT(4,19):"SO IST DAS" ::
    DISPLAY AT(6,19):"FELD, WENN"
1160 DISPLAY AT(8,19):"r STEINE" ::
    DISPLAY AT(10,19):"AUF rA"
1170 DISPLAY AT(12,19):"UND s AUF" ::
    DISPLAY AT(14,19):"FELD tC" ::
    DISPLAY AT (16,19):"LIEGEN."
1180 GOSUB 1250
1190 FOR N=1 TO 6 :: FOR K=1 TO 6 ::
    B(N,K)=W(N,K)-1 :: O(N,K)=1 :: NEXT K :: NEXT N
    :: GOSUB 680 :: GOSUB 800
1200 DISPLAY AT(4,19):"DIES IST" ::
    DISPLAY AT(6,19):"DIE" ::
    DISPLAY AT(8,19):"MAXIMALE"
1210 DISPLAY AT(10,19):"BELEGUNG" ::
    DISPLAY AT(12,19):"ALLER" ::
    DISPLAY AT(14,19):"FELDER."
1220 GOSUB 1250
1230 DISPLAY AT(12,4):"*** UND JETZT GEHT^S LOS ***"
1240 FOR N=1 TO 6 :: FOR K=1 TO 6 :: B(N,K)=0 ::
    O(N,K)=0 :: NEXT K :: NEXT N :: IO=0 ::
    CN$(1)="" :: CN$(2)=""
1250 REM SUB - INSTRUKTION SEITE WECHSELN
1260 DISPLAY AT(24,5):"DRUECKE EINE TASTE"
1270 CALL KEY(3,KEY,ST)::
    IF ST=0 THEN 1270 ELSE CALL CLEAR :: RETURN
1280 REM AUTOMATIK
1290 Q=0 :: U=Q :: FOR N=1 TO 6 :: FOR J=1 TO 6
1300 P(N,J)=0 ::
    IF O(N,J)=K AND B(N,J)=W(N,J)-1 THEN P(N,J)=1 ::
    Q=Q+1
1310 IF O(N,J)<>S THEN U=U+1
1320 NEXT J :: NEXT N :: IF Q=0 THEN 1470
1330 FOR N=1 TO 6 :: FOR J=1 TO 6 ::
    IF P(N,J)=0 THEN 1420
1340 OX=J :: FOR PX=N-1 TO 1 STEP -1 :: GOSUB 1590 ::
    IF B(PX,J)<>W(PX,J)-1 THEN PX=1
1350 NEXT PX :: IF N>5 THEN PX=1 :: GOTO 1380
1360 FOR PX=N+1 TO 6 :: GOSUB 1590 ::
    IF B(PX,J)<>W(PX,J)-1 THEN PX=6
```

```
1370 NEXT PX
1380 PX=N :: FOR OX=J TO 1 STEP -1 :: GOSUB 1590 ::
     IF B(N,OX)<>W(N,OX)-1 THEN OX=1
1390 NEXT OX :: IF J>5 THEN 1420
1400 FOR OX=J+1 TO 6 :: GOSUB 1590 ::
     IF B(N,OX)<>W(N,OX)-1 THEN OX=6
1410 NEXT OX
1420 NEXT J :: NEXT N
1430 M=0 :: FOR PX=1 TO 6 :: FOR OX=1 TO 6
1440 IF P(PX,OX)>M THEN M=P(PX,OX):: N=PX :: J=OX
1450 NEXT OX :: NEXT PX ::
     IF Q<>1 OR((N=1 OR N=6)AND(J=1 OR J=6)AND RN(2)=1)
     THEN 1570
1460 IF Q=1 AND P(N,J)>50 THEN 1570
1470 B3=0 :: IF RN(3)=1 OR U>33 THEN 1500
1480 FOR PX=1 TO 6 :: FOR OX=1 TO 6 ::
     IF O(PX,OX)=K THEN N=PX :: J=OX :: PX=6 :: OX=6
10-5 IT1H N1 RR IT1H Q1 RR YNHN 2625
2655 I{SI*7|RR V{SI*7|RR LC N*IWV|{E HOTI 2655
2625 S{5 RR CNS Q1{I$2 HN I$*I)7|RR
     LC N*Q1WV|)JE HOTI 2645
2635 LC F*PX,J)=W(PX,J)-1 THEN R=1
1530 NEXT PX :: FOR OX=J-1 TO J-(J<6)::
     IF O(N,OX)<>S THEN 1550
1540 IF B(N,OX)=W(N,OX)-1 THEN R=1
1550 NEXT OX :: IF R THEN B3=B3+1 ::
     IF B3<U THEN 1500
1560 GOTO 1570
1570 DISPLAY AT(20,18):CHR$(112+N);CHR$(J+64)::
     GOSUB 1670 :: X=J :: Y=N :: GOTO 540
1580 REM SUB - PUNKTE
1590 P(N,J)=P(N,J)+1 ::
     IF O(PX,OX)=S THEN P(N,J)=P(N,J)+50
1600 RETURN
1610 REM SUB - ERHOEHUNG NACHBARFELD
1620 IF O(Q,U)=S THEN D(S)=D(S)-1
1630 IF O(Q,U)<>K THEN D(K)=D(K)+1
1640 O(Q,U)=K :: RETURN
1650 REM SUB - MUSIK
1660 RESTORE 1740 :: FOR I=1 TO 31 ::
     READ T1,T2,T3,L :: CALL SOUND(L,T1,0,T2,6,T3,5)::
     NEXT I :: RESTORE :: RETURN
1670 CALL SOUND(150,240,2):: CALL SOUND(150,360,2)::
     CALL SOUND(300,120,2):: RETURN
1680 REM DATEN - ARRAY
1690 DATA 2,3,3,3,3,2
1700 REM DATEN - CHARACTER
1710 DATA 0000001F1F181818,000000FFFF,000000FFFF181818,
     000000F8F8181818,1818181818181818,
     1818181F1F181818,181818FFFF181818,
     181818F8F8181818
1720 DATA 1818181F1F,181818FFFF,181818F8F8
1730 REM DATEN - MUSIK
1740 DATA 740,496,147,400,494,494,147,200,587,494,147,
     200,740,494,294,200,587,494,294,200,494,494,294,
     400
1750 DATA 784,494,330,800,659,440,277,400,440,440,277,
     200,554,440,277,200,659,440,277,200,554,440,277,
     200
1760 DATA 440,440,277,400,740,440,294,800,659,494,196,
     200,740,494,196,200,659,494,196,200,587,494,196,
     200
1770 DATA 554,440,220,400,698,440,220,400,659,392,247,
     200,698,392,247,200,659,392,247,200,587,392,247,
     200
1780 DATA 440,392,220,400,587,349,220,400,554,330,220,
     200,587,330,220,200,659,392,220,200,587,392,220,
     200
1790 DATA 587,349,294,2400 ■ H.U.
```


Assembler-kurs ASEM-4



Mehr rausholen aus dem »Micro« mit schnelleren und leistungsfähigeren Maschinen-Programmen. —
Lernen Sie Assembler-Programmierung und Computer-Aufbau verstehen. **In direktem Bildschirm-Dialog** und begleitet von zahlreichen Übungsaufgaben, wird der Stoff in leicht verständlicher Form vorgestellt.

Zum Kurs gehören:

- **2 deutsche Handbücher (ca. 400 Seiten),**
- **Lösungsblätter und Referenzkarte,**
- **Software auf Diskette oder Cassette.**

Wählen Sie für Ihr System passend:

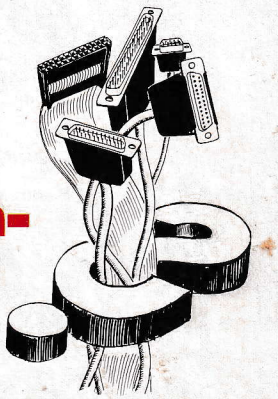
Cassette:	
CBM 3/40xx	98,-
CBM 80xx	98,-
VC-20 (16K)	98,-
C 64	98,-
ZX-81 (32K)	98,-
Spectrum (48K)	98,-
Genie I, II, III	98,-
Colour-Genie	98,-
TRS-80 I	98,-
ATARI	98,-
TI-99	98,-

Diskette:	
CBM 3/40xx (4040)	98,-
CBM 80xx (8050)	98,-
VC-20 (16K, VC-1541)	98,-
C 64 (VC-1541)	98,-
Apple II, IIe	98,-
Genie I, II, III	98,-
TRS-80 I (35-Track)	98,-
Colour-Genie	98,-
MS-DOS (IBM-PC u.v.a.)	148,-

CP/M, je nach Rechner-Typ: auf Diskette oder auf Cassette mit Cass.-Interface über V24-Schnittstelle **148,-**

. . . andere Systeme auf Anfrage!

Schnittstellen-Probleme?



Schnittstellen-Probleme schnell erkennen und beheben! Für viele Anwendungsfälle haben wir den passenden Interface-Tester oder Adapter:

RS 232 / V24 Schnittstellen-Tester, unterbrechen, kreuzen und überbrücken aller Datenleitungen (DIP-Switches / Steckfelder), Signal-Anzeige durch LEDs, BNC-Buchsen für Oszilloscope-Anschluß, ein Gerät für die Praxis. Komplett mit Kabeln und deutscher Anleitung **DM 548,-**

RS-232 / V24 Mini-Adapter, löst das bekannte Problem unterschiedlicher Pin-Belegungen; unterbrechen, vertauschen und brücken aller Datenleitungen, Anpassung an spez. Peripherie-Geräte in Sekunden.
DM 132,- (1 St.) **DM 114,-** (ab 3 St.) **DM 99,-** (ab 10 St.)

Adapter für andere Kabel-Systeme auf Anfrage!

Alle Preise incl. MwSt. **Versand per NN**

ING.-BÜRO WILKE

Postfach 1727 · 5100 Aachen 1 · Tel. 0241 / 3 06 81 + 87 02 08

SOFTWARE IM TEST

Flugsimulator

Diese Cassette beinhaltet einen Flugsimulator in EXTENDED BASIC und TI BASIC und wird mit Joysticks und Tastatur bedient.

Die Flugeigenschaften des simulierten Flugzeugs entsprechen in etwa denen eines Jumbo Jets: Der Flugsimulator reagiert auf Joystickanweisungen nur sehr langsam. Nun soll versucht werden, den Jet aus einer Höhe von 1300 Metern sicher auf dem Boden zu landen. Dabei müssen Landeklappen und Fahrwerk ausgefahren und der Instrumenten-Lande-System-Empfänger (ILSE) eingeschaltet werden. Anweisungen, wie zu fliegen ist, erhält man vom Landekurssender des Towers.

Auf dem Bildschirm sieht man dann schließlich das Cockpit des Flugzeugs,

einen künstlichen Horizont, einen Entfernungsanzeiger, den geflogenen Kompaßkurs, die momentane Flughöhe, die Geschwindigkeit und noch eine ganze Menge weiterer Anzeigen.

Nachteilig an diesem Spiel ist vor allem die fehlende Graphik. Die Mühe eines Landebahnentwurfs haben sich die Spielautoren einfach gespart. Nicht einmal Wolken tauchen auf. Übrig bleibt ein Spiel, das durch seine Komplexität, aber keineswegs durch seine Graphik oder seine Schnelligkeit Spielspaß vermittelt. Wer keine Ausdauer hat, sollte sich dieses Spiel allerdings nicht kaufen, denn Ausdauer braucht man, um die durch die Praxisnähe bedingte anfängliche Frustration zu überstehen, um nicht die Cassette in die Ecke zu schmeißen und sie nie wieder auch nur eines einzigen Blickes zu würdigen.

Graphik:

Sound:

Spielidee:

Testergebnis:

Sympathie

Unseres Wissens haben wir nie einen Eid abgelegt, Bücher objektiv zu beurteilen. Schließlich sind wir selbst User und es wäre schon komisch, wenn wir das vergessen würden. Und so bricht uns kein Zacken aus der Krone, wenn wir sagen: dieses Buch für den TI 99/4A gefällt uns einfach, weil der Schreibstil witzig und symphatisch ist. Echter Lesespaß.

Inhaltlich bietet es neben kurzen Abschnitten über "Farben und Grafik", "Töne und Akkorde" sowie "Kniffe und Tricks" eine Menge ausgefuchster Spiele. 192 Seiten Listings mit Einleitung und Erklärung zu jedem Spiel (den Autor hat die lockere Schreibe bis ans Ende des Buches zum Glück nicht verlassen). Die Listings sind auf 28 Zeichen pro Zeile formatiert, was günstig für das Abtippen ist. Sehr gut gefiel uns, daß parallel zu den Listings laufende Dokumentation steht.

Als Spiele finden sich hier, um nur einige zu nennen: FORMEL-I, LABYRINTH-LAUF, SUPERMIND (nun, es hat ja keiner gesagt, daß alles neu ist), SENSOCOMP, SQUASH und TENNIS, ELIZA als Beispiel von AI (Künstlicher Intelligenz) und SCHWARZES MEMORY zur Illustration von AD = Artificial Dumbness (Künstlicher Dummheit). Und vieles mehr.

Zu jedem Listing gibt es Anregungen, das Programm zu ändern, getreu dem Motto, daß zumindest der Fortgeschrittene angeregt werde, seiner Phantasie freien Lauf zu lassen.

"Nur zu, Freunde! Diesen kleinen Riesen TI 99/4A gilt es, an seine Grenzen zu führen. Viel Spaß". Dem können wir nichts hinzufügen.

Guido Pahlberg
IWT-Verlag München
ISBN 3-88322-045-0
217 Seiten
39,— DM

TI 99/4A Farbe, Grafik, Ton, Spiele (ist als Programm-Cassette ebenfalls lieferbar)

ThoK

Endlich sind die ersten Atari-Moduln auch für den TI99/4A zu bekommen. Die Firma Atari hat uns die beiden ersten adaptierten Steck-Moduln

ZU TI TELEFONBUCH
Heft 5/84, Seite 76

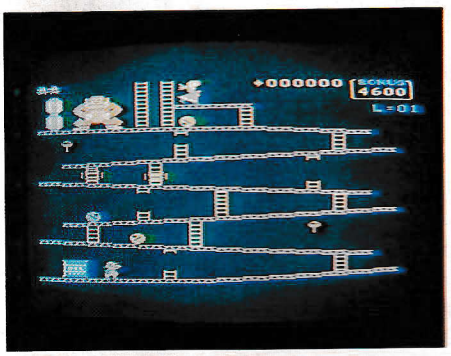
Listing-Zeile 130 ist nur mit Trick in den Rechner zu bringen!

Nach der Meldung "Zeile voll" (akustisches Signal) ENTER drücken, Editier-Modus verlassen, Zeile 130 neu editieren (z.B. mit NUM 130).

Eingabe fortsetzen (natürlich am Ende der bereits eingegebenen Zeile), bis erneutes Signal kommt, dann den Ablauf wiederholen.

DIE ZEILE LÄSST SICH NUR SO EINGEBEN!

zugeschickt: Donkey Kong und Defender. Beide Spiele dürften aus Spielhallen oder von Atari- bzw. Commodore-Rechnern vielen von euch bekannt sein. Für alle, die noch nichts davon gehört haben, hier eine kurze Beschreibung:



Donkey Kong

Marios Freundin ist von dem furchterregenden Monster-Affen Donkey Kong gefangen und auf ein Baugerüst verschleppt worden. Mario muß jetzt zu seiner Freundin hinaufklettern, um sie zu retten. Aber das gefällt Donkey Kong gar nicht, und deshalb wirft er mit allerlei Gegenständen um sich: mit Fässern, Sandeimern und Flammwesen, die den armen Mario auf seinem Weg nach oben attackieren. Jedesmal, wenn es ihm gelingt, über eines dieser Hindernisse hinwegzuspringen, werden ihm zur Belohnung Punkte gutgeschrieben.

Die bekannte, ansprechende Graphik des Spiels ist von Atari erfolgreich auf den TI übertragen worden. Nur der Sound, besonders die Start-Melodie, kann einem schnell auf die Nerven gehen. Aber da helfen die — ebenfalls bekannten — Möglichkeiten eines Lautstärke-Reglers.

Graphik: ☺

Sound: ☺

Spielidee: ☺

Testergebnis:

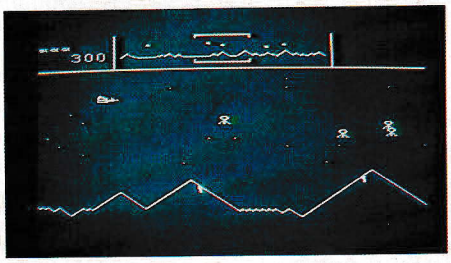
☺

Defender

Eines der ältesten 'Baller-Spiele' überhaupt. Einziges Ziel: schießen, schießen, schießen. Und natürlich treffen! Wie es in der Anleitung steht: die „totale Vernichtung" des Gegners wird angestrebt. Der ist aber auch nicht faul, versucht er doch, mit Spezial-Schiffen die 'Humanoiden' des Verteidigers zu kidnappen und als Mutanten auf seiner Seite wieder in den Kampf zu schicken. Mal ganz davon abgesehen, daß die Angreifer auf den Verteidiger blindwütig schießen, was das Zeug hält.

Hat man erfolgreich alle Feinde vernichtet — rollt schon die nächste Angriffswelle an.

Die sowieso recht einfache Graphik ist problemlos an das TI-System adaptiert worden. Die hektischen Bewegungen des Defender-Raumschiffes gehen



aber reichlich auf die Augen. Großartige Variationsmöglichkeiten gibt es nicht. Auch der Sound ist nach einfachstem Muster gestrickt: man hat sich auf das übliche 'Weltraum-Rauschen und -Krachen' beschränkt.

Jedes dieser beiden Spiele kostet ca. 100,— Mark. Dafür bekommt man jeweils das Steckmodul mit einer ausführlichen deutschen Anleitung, in der sowohl die Inbetriebnahme des Moduls als auch das Spiel selbst genau beschrieben werden. Kleine Fehler, wie z.B. die Anweisung, die Joysticks an der rechten Seite des Rechners anzuschließen, muß man großzügig überlesen.

Übrigens: beide Spiele lassen sich nur mit Joysticks betreiben! Dafür allerdings auch zu zweit. Bleibt nur das Problem, wo man vernünftige Steuerknüppel herbekommt, sofern man keine TI-Original-Joysticks besitzt.

Einen Adapter für TI-fremde Joysticks haben wir in dieser Ausgabe beschrieben. Aber trotz dieser Möglichkeit zur Selbsthilfe ist es ein schwaches Bild, daß professionelle Software nach wie vor 'Peripherien' verlangt — und seien es nur Joysticks.

Graphik: ☹

Sound: ☹

Spielidee: ☹

Testergebnis:

☹

to be AND OR NOT to be!

Präzise Aussagen sind nötig, wenn man mit dem Computer plauschen will. Na und? Oder oder? Oder nicht?

In dieser Ausgabe von COMPUTER Praxis wollen wir die TI-User mit der Möglichkeit vertraut machen, logische Funktionen — wie AND, OR und NOT — auch ohne Zuhilfenahme von Ex-Basic zu realisieren.

Was sind logische Funktionen? Sie ermöglichen dem Programmierer, mit IF ... THEN-Befehlen bedingte, d.h. von ganz bestimmten Bedingungen abhängige, Programmsprünge zu anderen Programmteilen zu machen. Bei IF...THEN-Befehlen erfolgt ein Sprung nur dann, wenn der zwischen IF und THEN stehende Ausdruck "wahr" ist. Dieser Ausdruck kann aus einer Vielzahl von logischen Verknüpfungen bestehen. Die wichtigsten logischen Verknüpfungselemente sind AND, OR und NOT.

Sag' die Wahrheit!

Wenden wir uns zunächst dem AND zu. Werden zwei Elemente, z.B. A und B, mit AND verknüpft, so ist das Ergebnis nur dann wahr, wenn sowohl A als auch B wahr ist. Das klingt kompliziert, ist es aber gar nicht, wie ihr gleich an einem einfachen Beispiel sehen könnt. Nehmen wir einmal an, A und B wären zwei Schalter und das wahre Ergebnis soll eine brennende Glühlampe sein. Das bedeutet für die Verknüpfung AND, daß die Lampe dann leuchten muß, wenn sowohl Schalter A als auch Schalter B eingeschaltet, sprich wahr sind. Aus Abb.1 könnt ihr ersehen, daß die Schalter hintereinander geschaltet werden müssen, um unsere AND-Schaltung zu erhalten. Wird nur ein Schalter — entweder A oder B betätigt —, so fließt kein Strom zur Lampe. Das Gleiche geschieht, wenn wir gar keinen Schalter drücken. Nur bei Betätigung beider Schalter kann Strom zu unserer Lampe fließen. Das können wir auch mit einer sogenannten Wahrheitstabelle allgemeiner darstellen. Hier könnt ihr in Abb.2 sehen, daß unser Ergebnis C nur dann wahr ist, wenn auch A und B wahr sind. In einer solchen Tabelle wird, wie ihr sicherlich schon bemerkt habt, für jede Verküpfungsmöglichkeit das entsprechende Ergebnis angegeben.

Abb. 2

Ereignis		Ergebnis
A	B	C
falsch	falsch	falsch
wahr	falsch	falsch
falsch	wahr	falsch
wahr	wahr	wahr

Abb. 1

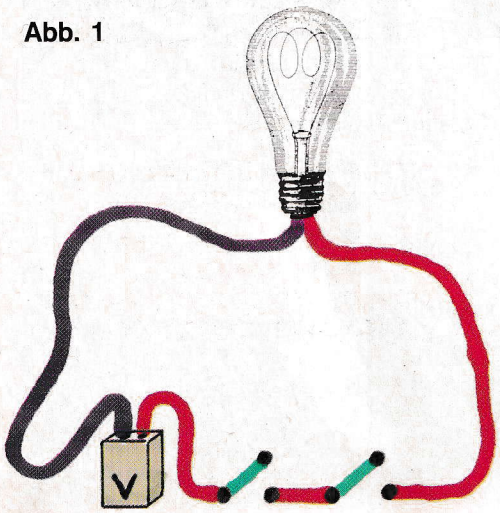
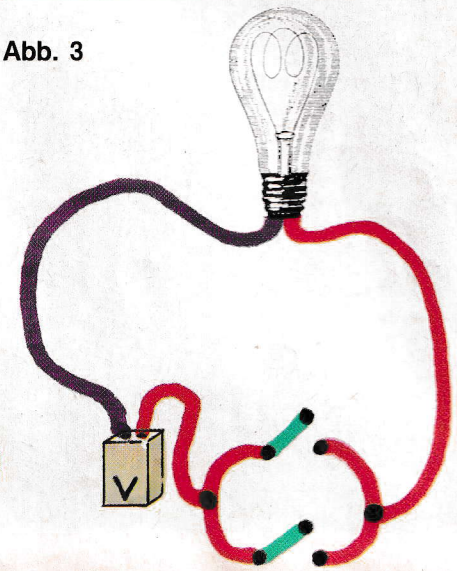


Abb. 3



Die OR-Verkünpfung wird zu Recht auch als Entweder-Oder-Verküpfung bezeichnet. D.h. werden zwei Ereignisse A und B durch OR verknüpft, so ist das Ergebnis C dann wahr, wenn entweder A oder B oder beide wahr sind. Für unser 'Lampenbeispiel' sieht die Schaltung wie in Abb. 3 gezeigt aus. Wird einer der parallel geschalteten Schalter oder beide betätigt, fließt der Strom wieder zu unserer Lampe. Eine Wahrheitstabelle für OR zeigt Abb.4.

Abb. 4

Ereignis		Ergebnis
A	B	C
falsch	falsch	falsch
wahr	falsch	wahr
falsch	wahr	wahr
wahr	wahr	wahr

Die NOT-Verknüpfung stellt alles im wahrsten Sinne des Wortes auf den Kopf: Ist ein Ereignis A wahr, so soll das Ergebnis C falsch sein und umgekehrt. Wird also ein Schalter A betätigt, muß unsere brennende Lampe ausgehen. Eine Schaltung dafür solltet ihr euch einmal selbst überlegen. Die Wahrheitstabelle ist in Abb.5 zu finden.

Abb. 5

Ereignis	Ergebnis
A	C
falsch	wahr
wahr	falsch

Nach soviel Theorie fragt ihr euch sicherlich: Wofür brauchen wir eigentlich diese logischen Verknüpfungen in unserem Programm? Das ist ganz einfach zu beantworten. Nehmen wir einmal an, ihr wollt ein Spiel programmieren, in dem es für den Spieler fünf Männchen

gibt, die von mehreren Monstern verfolgt werden. Sinn des Spiels ist es, sich möglichst lange nicht erwischen zu lassen. Nun taucht beim Programmieren sicher die Frage auf, wann ist das Spiel beendet? Doch, sobald das fünfte Männchen von einem Monster erwischt wird! Um den Sprung zum Programmende zu programmieren, müssen genau zwei Ereignisse eintreten, die den Sprung vollziehen. Erstens: Mein Männchen muß erwischt werden, und zweitens: Es muß mein letztes Männchen sein. Ganz Schlaue werden jetzt bemerkt haben, daß es sich hier um eine AND-Verknüpfung handeln muß. Denn sowohl Ereignis A — mein Männchen wird geschnappt — als auch Ereignis B — es ist mein letztes Männchen — müssen erfüllt, sprich wahr sein, damit das Ergebnis C — Sprung zum Ende des Programms — auch wahr wird.

Auch eine OR-Verknüpfung ist in unserem Programm denkbar. Es kann nämlich eintreten, daß unser Männchen entweder von einem oder von mehreren Monstern gleichzeitig einkassiert wird. In jedem Fall gilt unser Männchen als erwischt, und ein neues Männchen muß ins Spiel gebracht werden. Der Sprung

FUNKTIONIERT OHNE FRUST: KOMFORTABLE SOFTWARE VON DYNAMICS.

Weil es nicht nur Computer-Freaks gibt, bietet Dynamics anwender-freundliche Software – komfortabel, menuegesteuert über den Bildschirm.

Wer also auf seinem Commodore C64, dem TI99/4A oder den Atari 400/800- oder XL-Modellen nicht lange programmieren will oder null Bock auf Listings hat, der geht ins Fachgeschäft oder Kaufhaus.
Dort gibt es das Dynamics **„Adressen-Archiv"**, das **„Video-"**, **„Bücher-"** und **„Schallplatten-Archiv"**: Programme, die Ordnung halten – einfach und flink. Da gibt es neu den **„Vokabel-Trainer"**, der Lernen zum Vergnügen macht. Fragen Sie danach!

Wer mehr will, sollte sich die Dynamics Arbeits-Programme zeigen lassen. Aber auch **„Word-Proc"**, **„Tool Pack I** und **II"**, **„Calculator"** oder **„Data-Bank"** bleiben immer anwender-freundlich.

Gut verständliche Bedienungsanleitungen (keine Wälzer!), logischer Aufbau und einfache Anwendung garantieren, daß computern mit Dynamics Spaß macht!

Also: auf ins Fachgeschäft, hin zum Kaufhaus. Weil Dynamics Software einfach komfortabel ist.
Ausprobieren!

DYNAMICS®

COMPUTER-SOFTWARE UND COMPUTER-ZUBEHÖR.

FÜR ATARI 400/800 UND 600 XL/800 XL, TEXAS INSTRUMENTS TI 99/4A UND COMMODORE C 64.

Dynamics-Software gibt es auf Computer-Cassette oder Diskette.

Dynamics Marketing GmbH,
Große Bäckerstraße 11,
2000 Hamburg 1.

wird demnach dann wahr, wenn unser Spieler entweder von einem oder mehreren geschnappt wird.

Eine NOT-Bedingung hätten wir z.B. dann vorliegen, wenn ein Sprung zum weiteren Programmablauf immer dann erfolgen soll, falls unser kräftiger Spieler nicht auf frischer Tat ertappt wurde.

Das Ex-BASIC und die meisten anderen BASICs erlauben die direkte Benutzung der Vokabeln AND, OR und NOT. Das ist beim TI-BASIC nicht so. Deshalb müssen wir uns an dieser Stelle etwas einfallen lassen.

Zuallererst wollen wir uns ansehen, was beim TI "wahr" und "falsch" bedeutet. Dazu solltet ihr einmal nachstehendes Listing eingeben.

```
100 INPUT A
110 IF A THEN 200
120 GOTO 300
200 PRINT "WAHR"
210 GOTO 100
300 PRINT "FALSCH"
310 GOTO 100
```

Bei diesem kleinen Programm könnt ihr verschiedene Werte für A eingeben und bekommt je nach Eingabe eine wahre Aussage oder eine falsche. Ihr habt sicherlich bemerkt, daß alles ungleich 0 "wahr" ist, also nur 0 für "falsch" und alle andere Werte — ob positiv oder negativ — in jedem Fall für "wahr" stehen.

Monströsitäten

Nun wollen wir zu unserem Männchenbeispiel zurückkehren. Wir hatten festgestellt, daß unser Sprung zum Programmende dann ausgeführt werden soll, wenn wir erwischt wurden, und unser letzter Spieler der Betroffene war. Das kann man in EX-BASIC, wie in folgendem Listing ersichtlich, leicht bewerkstelligen:

```
100 IF (E=1) AND (M=5) THEN
    1000
110 REM HIER LAEUFT UNSER
    PROGRAMM WEITER
120 REM WENN WIR NICHT ER-
    WISCHT WURDEN ODER
130 REM WIR ZWAR ERWISCHT
    WURDEN ABER NOCH
140 REM NICHT MIT UNSEREM
    LETZTEN MAENNCHEN
1000 REM HIER BEGINNT UNSERE
     ENDE-ROUTINE
1010 REM UNSER LETZTES MAENN-
     CHEN WURDE ERWISCHT
```

Und zwar steht E für erwischt und wird vorher von uns immer dann auf eins gesetzt, wenn wir geschnappt wurden. M zeigt an, welches Männchen momentan im Spiel ist. Immer wenn E=1 und(!) M=5 ist, verzweigt unser Programm zur Zeile 1000. Nun kennt das normale TI-BASIC das Schlüsselwort "AND" aber nicht. Wir müssen uns also etwas einfallen lassen! Die Lösung ist ganz einfach. Folgende Programmzeile löst all unsere Probleme:

```
100 IF (E=1)*(M=5) THEN 1000
```

Statt des AND ein Multiplikationszeichen? Richtig! Die Gründe dafür sind sehr einfach zu verstehen. Tippt einmal PRINT 1=1, was sicherlich richtig ist, ein und ihr werdet eine -1 für "wahr" erhalten. Bei PRINT 1=3, das führt logischerweise zu einem "falschen" Ergebnis, erhaltet ihr den Wert 0. D.h., ist ein Vergleich richtig, liefert der TI uns eine -1, und ist er falsch, eine 0. Ist aber in unserer obigen Zeile ein Vergleich falsch, so ist dieser gleich 0. Eine Multiplikation mit einer Null ergibt unabhängig davon, womit ich sie multipliziere, immer eine Null als Ergebnis, und diese steht, wie wir gesehen haben, für "falsch". Nur wenn beide Vergleiche also richtig sind (=-1), wird unser Ergebnis auch "wahr" sein (-1 multipliziert mit -1 ergibt 1. 1 versteht unser TI aber auch als "wahr"). Der Vergleich mit unserer Wahrheitstabelle zeigt uns, daß wir hier anscheinend richtig programmiert haben.

Jetzt zur OR-Bedingung. Wir hatten in unserem Beispiel gesagt, daß wir dann verzweigen wollten, wenn uns eins oder mehrere Monster erwischt haben. Nehmen wir an, daß in unserem Programm für die drei Monster, die uns jagen, jeweils eine Abfrage erfolgt, ob sie uns ertappt haben. Ist das der Fall, dann wird für Monster 1 die Variable M1 gleich 1 gesetzt, für Monster 2 die Variable M2 usw. Unser EX-BASIC-Programm sieht unter diesen Voraussetzungen so aus:

```
100 IF (M1=1) OR (M2=1) OR
    (M3=1) THEN 1000
110 REM WIR WURDEN NICHT
    ERWISCHT UND
120 REM UNSER PROGRAMM
    LAEUFT WEITER
    .
    .
    .
1000 REM HIER BEGINNT DIE
     "WIR-WURDEN-ERWISCHT
     ROUTINE"
```

Wir sehen: Ist auch nur eine der Variablen M1 bis M3 gleich 1, springt unser Programm in Zeile 1000. In TI-BASIC müssen wir uns wieder behelfen. Wir nehmen statt der Multiplikation einfach die Addition, und schon funktioniert unser Befehl auch im normalen BASIC:

```
100 IF (M1=1)+(M2=1)+(M3=1)
    THEN 1000
```

Keine Not mit NOT

Das funktioniert folgendermaßen. Wird nur ein einzelner Vergleich wahr, so erhält dieser und damit der ganze Ausdruck den Wert -1. Ertappen uns mehrere Monster, erhalten wir die Werte -2 oder-3. -1 bis -3 sind jedoch alles "wahre" Werte. Nur wenn uns keiner erwischt, ist auch die Addition dieser "Nullen" eine Null und das ist eine "falsche" Aussage. Darum erfolgt in diesem Fall auch kein Sprung.

Last but not least die NOT-Verknüpfung. Diese wollen wir durch Probieren studieren:

```
100 INPUT A
110 IF (A=0)THEN 200
120 GOTO 300
200 PRINT "WAHR"
210 GOTO 100
300 PRINT "FALSCH"
310 GOTO 100
```

Geben wir für A "wahre" Werte (positive oder negative Zahlen) ein, ergibt das ein falsches Ergebnis. Wird A=0 wie "falsch" gesetzt, bekommen wir "wahr" auf unseren Bildschirm. Warum? Wir vergleichen unsere Eingabewerte mit 0, ist dieser Vergleich "wahr", dann müssen wir vorher eine "falsche" Zahl (=0) eingegeben haben und entsprechend umgekehrt.

Mit diesem Rüstzeug versehen, könnt ihr nun eure Programme hoffentlich besser gestalten. ■

J.P.P.

Parallel und seriell

2. Folge
Schnittstellen sind nicht nur zum Drucken da. Auch Computer finden sich zu einem Kaffeekränzchen ein, wenn man sie Verbindungen knüpfen läßt.

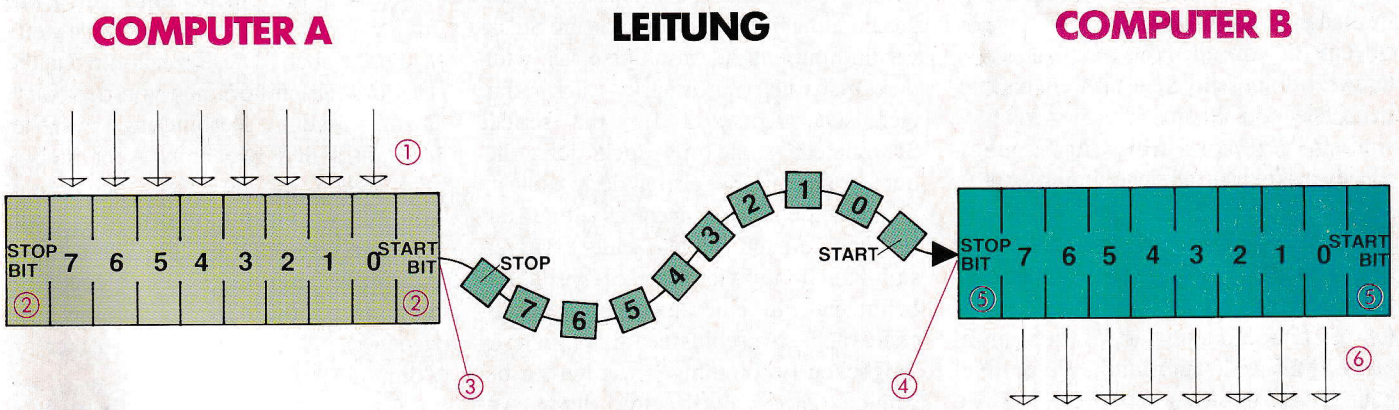


Bild 1: **Der Ablauf der seriellen Übertragung**

1. Computer A schreibt ein Zeichen (= Byte) in den Schnittstellenbaustein.
2. Dem Zeichen werden Start- und Stop-Bits hinzugefügt.
3. Das Zeichen wird Bit für Bit auf die Leitung geschoben.
4. Am anderen Ende der Leitung wird es wieder in ein Register geschoben.
5. Start- und Stop-Bits werden entfernt.
6. Computer B kann das Zeichen lesen.

In der ersten Folge unserer Serie über Schnittstellen am Computer haben wir gesehen, wie sich der Computer mit einem Drucker 'unterhalten' kann. Allerdings ist diese Unterhaltung recht einseitig, denn hier 'redet' nur einer — der Computer, der Zeichen an den Drucker ausgibt. Interessanter — und auch ein wenig komplizierter — wird es, wenn wir ein Terminal anschließen oder zwei Computer miteinander verbinden wollen. Dann gehen nämlich Daten in beiden Richtungen hin und her, der Computer muß senden und empfangen können. Dazu benutzt man meistens serielle Schnittstellen — und wie die funktionieren, wollen wir euch in dieser Folge zeigen.

Im Gänsemarsch

Seriell... was bedeutet das eigentlich? Wir könnten statt 'seriell' auch 'in Serie' oder einfach 'nacheinander' sagen. Diese Worte beschreiben die Funktionsweise der Schnittstelle recht gut, denn bei der seriellen Übertragung werden alle Bits nacheinander auf die Reise zum Empfänger geschickt. Der Vorteil dieses Verfahrens ist klar: Statt acht Leitungen (je eine für jedes Bit eines Bytes) zu verlegen, braucht man nur eine Leitung. Dafür ist der technische Aufwand größer, denn der Computer selbst arbeitet ja intern immer mit acht Datenleitungen (dem 'Datenbus'). Nehmen wir an, wir wollen ein Zeichen (also ein Byte) von Computer A zu Computer B schicken. Wie bekommt man dann die Bits von den acht Leitungen im Rechner A nacheinander auf die eine Übertragunsleitung zu Computer B?

Dafür gibt es besondere IC's die mit ACIA, UART, SIO oder anderen phantasievollen Abkürzungen bezeichnet werden. Ihre Hauptbestandteile sind zwei Schieberegister — eines zum Empfangen und eines zum Senden. In diese Register kann der Computer schreiben (z.B. mit "POKE") oder lesen ("PEEK"), wie in jede gewöhnliche Stelle im Speicher. Das Besondere ist nun, daß ein Byte, das man per Programm in solch ein Register geschrieben hat, automatisch über einen extra Ausgang ausgegeben wird. Dabei wrden die einzelnen Bits nacheinander hinausgeschoben — daher der Name 'Schieberegister'.

An diesen Ausgang schließen wir natürlich unsere Übertragungsleitung von Computer A zu Computer B an. Am anderen Ende der Leitung gelangen die Bits wieder an ein Schieberegister — aber dieses Mal an einen Eingang. Durch diesen Eingang werden die Bits in das Register hineingeschoben. Ist ein Byte komplett, kann es vom Programm im Computer B gelesen werden. Im Bild 1 haben wir den Ablauf der seriellen Übertragung für euch aufgezeichnet.

Die oben genannten IC's — ACIA, UART und so weiter —, die die Schieberegister enthalten, können aber noch mehr. Sie fügen vor dem Senden eines Bytes noch besondere Bits hinzu — ein Startbit vor jedem Byte und ein Stopbit nach jedem Byte. Der empfangende Computer hat nichts besseres zu tun, als diese Bits gleich wieder abzutrennen und

"wegzuwerfen", aber Start- und Stopbit helfen ihm, mehrere aufeinanderfolgende Bytes zu erkennen und sie auch auseinander zu halten.

Die einzige Bedingung, die wir erfüllen müssen, ist, daß sich Sender und Empfänger über die Geschwindigkeit einig sind, mit der die einzelnen Bits übertragen werden. Wenn der Sender zum Beispiel 1000 Bits pro Sekunde ausgibt, muß der Empfänger natürlich genauso oft ein Bit von der Übertragungsleitung lesen und in sein Register hineinschieben. Die Anzahl der Bit pro Sekunde sind also ein Maß für die Übertragungsgeschwindigkeit. Üblich sind 50, 110, 300, 600, 1200, 2400, 4800, 9600 und 19200 Bit pro Sekunde. An dieser Stelle sind nun zwei Fachausdrücke fällig. Für 'Bit pro Sekunde' sagt man 'Baud' (sprich "Bood") und statt Übertragungsgeschwindigkeit sagt man in diesem Fall 'Baudrate' (sprich "Boodreet"). Also zum Beispiel: "Bei den meisten Computer läßt sich die Baudrate zwischen 300 und 9600 Baud einstellen ..." Das ist computerchinesisch, aber ihr wißt ja nun, was es bedeutet.

Und eine kleine Denksportaufgabe dazu: wieviele Zeichen kann Computer A pro Sekunde an Computer B senden, wenn beide auf 9600 Baud eingestellt sind? Und wieviele Zeichen kann A senden, wenn A auf 9600, B aber auf 4800 Baud eingestellt ist? Na?

Wie oben gesagt, besteht jedes Zeichen aus acht Bit, dazu kommen noch ein Start- und ein Stop-Bit. Macht 10 Bit pro Zeichen. Bei 9600 Baud (= Bit pro Sekunde) sind es also 960 Zeichen.

Im zweiten Fall ist es ganz egal, wieviele Zeichen Computer A schickt, weil er nicht mit der gleichen Baudrate wie Computer B arbeitet: es wird kein Zeichen richtig ankommen ...

Was ist V.24?

Nachdem wir die technischen Grundlagen der seriellen Schnittstelle besprochen haben, werden die Leser der ersten Folge schon ahnen, was nun kommt: die Norm. Ihr erinnert Euch doch — Normen ermöglichen es, Geräte unterschiedlicher Hersteller problemlos miteinander zu verbinden. Die serielle Schnittstelle ist sogar so wichtig, daß sich gleich drei Insitutionen mit ihr beschäftigt haben und es folglich drei Normen gibt: V.24, RS-232-C, DIN 66020. Zum Glück sind die Unterschiede sehr klein und für unsere Anwendungen können wir sogar sagen, daß die drei Normen gleich sind.

Gleich sind auf jeden Fall die verwendeten Buchsen und Stecker, die sogenannten Miniatur-D-Steckverbindungen, die Ihr in Bild 2 seht. So eine Steckverbindung hat 25 Anschlüsse. Da wundert man sich natürlich — der ganze technische Aufwand für die serielle Schnittstelle sollte uns doch dazu dienen, weniger Kabel als bei der parallelen Schnittstelle zu verlegen! Richtig, tatsächlich müssen wir für eine 'Sparversion' auch nur vier der 25 Anschlüsse benutzen. Für eine 'Luxusversion' reichen acht Anschlüsse. Alle anderen möglichen Verbindungen werden für besondere Zwecke verwandt, deren Anwendung für uns nicht in Frage kommt.

Trotzdem, bisher haben wir nur von einer einzigen Leitung gesprochen. In unserem Beispiel war das die Übertragungsleitung von Computer A zu Computer B. Und wenn nun Computer B etwas an A senden möchte? Dafür nehmen wir eine zweite Leitung. Dazu kommen natürlich noch eine Masseleitung (sonst geht gar nichts) und eine 'Schutzerde', die die Chassis der beiden Geräte miteinander verbindet — zur Sicherheit. Habt ihr mitgezählt? Das waren schon die vier Leitungen unserer V.24-Schnittstelle in Sparversion.

Die Luxus-Version

Durch vier weitere Leitungen unterscheiden sich Spar- und Luxus-Version unserer seriellen Schnittstelle. Luxus bedeutet hier aber nicht unnötigen Komfort, vielmehr geht es um wichtige weitere Funktionen. Die Bezeichnungen dieser Funktionen kommen von der häufig anzutreffenden Verbindung zwischen Terminals und Modems. Aber am besten stellen wir uns hier noch einmal vor, daß wir einen Computer mit einem Drucker verbinden wollen. Dann haben die vier weiteren Leitungen folgende Bedeutungen (die Anschlußnummern beziehen sich jeweils auf den Stecker am Computer):

- ▷ Anschluß 20, Terminal betriebsbereit: der Computer teilt dem Drucker mit, daß er betriebsbereit ist. Der Drucker wird Daten, die er empfängt, nicht drucken, ehe dieses Signal nicht aktiv ist.
- ▷ Anschluß 6, Betriebsbereitschaft: ist das Gegenstück zu Anschluß 20. Der

Bild 2:
25-polige Miniatur-D-Steckverbindung

Anschluß	deutsche Beschreibung	englische Beschreibung	Abkürzung
1 —	Schutzerde	protective ground	
2 →	Sendedaten	transmit data	TD
3 ←	Empfangsdaten	receive data	RD
4 →	Sendeaufforderung	request to send	RTS
5 ←	Sendebereitschaft	clear to send	CTS
6 ←	Betriebsbereitschaft	data set ready	DSR
7 —	Masse	ground	
20 →	Terminal betriebsbereit	data terminal ready	DTR

Bild 3:
Übersicht über alle wichtigen Signale einer V.24 bzw. RS-232-C Schnittstelle
→ bedeutet: Ausgang bei einer Datenübertragungseinrichtung,
Eingang bei einer Datenendeinrichtung.
← bedeutet: Eingang bei einer Datenendeinrichtung,
Ausgang bei einer Datenübertragungseinrichtung.
Die Begriffe "Datenübertragungs-" und "Datenendeinrichtung" erläutern wir in der nächsten Folge!

Der Drucker teilt dem Computer mit, daß er betriebsbereit ist. Vorher sollte der Computer keine Daten senden — sie würden verloren gehen.

▷ Anschluß 4, Sendeteil einschalten: der Computer teilt mit, daß er Daten senden will.

▷ Anschluß 5, Sendebereitschaft: das Gegenstück zu Anschluß 4: der Drucker zeigt an, daß er Daten aufnehmen kann.

Verwirrend? Auf den ersten Blick erscheinen die Funktionen der Anschlüsse 20,6 und 4,5 sehr ähnlich. Die Betriebsbereitschaft (20,6) ist ein eher übergeordnetes Signal; meistens wird es schon aktiv, wenn das Gerät eingeschaltet wird. Der Drucker könnte es aber durchaus benutzen, um anzuzeigen, daß er nicht betriebsbereit ist, weil beispielsweise kein Papier eingelegt ist. Die Anschlüsse 4 und 5 regeln den Datenaustausch mehr im Detail.

Übrigens findet man in der Literatur und besonders auch in den Unterlagen der Heimcomputer-Hersteller meistens die englischen Bezeichnungen für die verschiedenen Funktionen der seriellen Schnittstelle. Deshalb haben wir in der Tabelle in Bild 3 die deutschen und englischen Bezeichnungen und die gebräuchlichen Abkürzungen für euch einmal zusammengestellt.

Und noch etwas Technik

Die Centronics-Schnittstelle in der ersten Folge arbeitete mit ganz 'normalen' Spannungen zwischen 0 und 5 Volt (den sogenannten 'TTL-Pegeln'). Aber nicht so die V.24-Schnittstelle! Hier werden die Spannungen, mit denen die integrierten Schaltungen in unseren Computern normalerweise arbeiten, noch invertiert und verstärkt. An den Ausgängen liegt dann eine Spannung zwischen +3 und +15 Volt für einen niedrigen (!) Pegel oder eine Spannung zwischen -3 und -15 Volt für einen hohen Pegel. An den Eingängen sind natürlich entgegengesetzt arbeitende Verstärker vorhanden, die die hohen Spannungen wieder in den TTL-Bereich umsetzen.

Wozu der Aufwand? Damit die Übertragung auch über längere Strecken fehlerlos funktioniert! Bei den eben genannten Spannungen liegt ein großer Abstand zwischen einem hohen und einem niedrigen Pegel — mindestens sechs Volt. Dadurch wird die Übertragung unempfindlich gegen Störungen aus der Umwelt. Außerdem ist der Spannungsbereich, innerhalb dessen ein Signal noch richtig erkannt wird, sehr groß. Selbst wenn zum Beispiel am Ausgang des Computers +12 Volt anliegen, und am Drucker kommen nur noch +9 Volt an — macht nichts, das Signal ist immer noch in Ordnung. Außerdem haben die zusätzlichen Verstärker noch eine Reihe angenehmer Eigenschaften. Vor allem sind sie kurzschlußfest.

Chips in Gefahr!

Ehe nun massenweise Heimcomputer zur Reparatur gebracht werden müssen, wollen wir eines noch ganz klar machen: eine serielle Schnittstelle und eine V.24 (oder RS-232-C)-Schnittstelle sind nicht das gleiche. Eine serielle Schnittstelle an sich tut nichts weiter, als eine Folge von Bits auszugeben oder einzulesen. Die integrierten Schaltungen, die das machen, arbeiten mit ganz normalen Spannungen, also im Bereich zwischen 0 und 5 Volt. Erst, wenn eine serielle Schnittstelle einer der genannten Normen entspricht, sind die zusätzlichen Verstärker an den Ein- und Ausgängen vorhanden, die Spannungen zwischen 0 und 5 Volt auf solche zwischen -15 und +15 Volt umsetzen (und umgekehrt). Hütet euch also davor, euren Heimcomputer, dessen serielle Schnittstelle nicht einer dieser Normen entspricht, an einen Drucker oder irgendein anderes Gerät mit einer V.24 (oder RS-232-C)-Schnittstelle anzuschließen! Bei diesem Versuch würde mit Sicherheit einiges kaputt gehen!

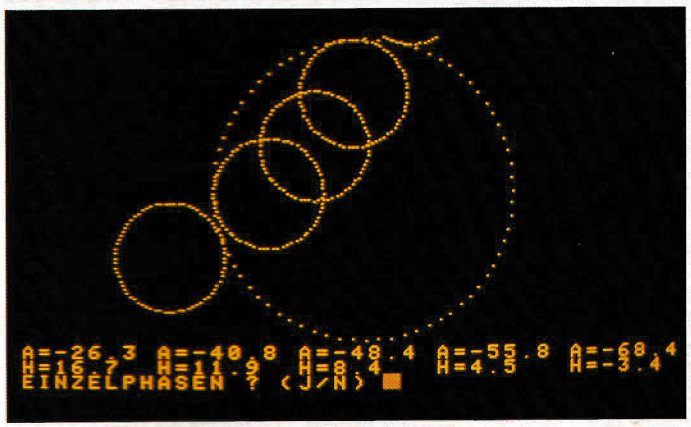
Wenn ihr nun die ersten beiden Folgen unserer Serie über Schnittstellen vergleicht — also die Centronics- und die V.24-Schnittstelle —, dann ging es diesmal entschieden komplizierter zu, als beim ersten Mal. Aber dafür bietet die serielle Schnittstelle auch mehr Möglichkeiten. Sie hat weniger Leitungen und kann trotzdem Daten in zwei Richtungen übertragen. Sie arbeitet mit höheren Spannungen und deswegen dürfen die Übertragungsleitungen ohne weiteres 30 Meter lang sein. Außerdem ist sie mit Übertragungsraten bis 19200 Baud auch ganz schön schnell.

Im nächsten Heft von COMPUTER Praxis werden wir auch ganz schön praktisch. Die dritte Folge gibt nämlich unter dem Titel "Wenn's doch nicht läuft" viele Tips, um Geräte mit seriellen Schnittstellen dann auch untereinander zu verbinden. ■

Klaus Liebenwald

Himmelsereignisse

Für die meisten User ist der Computer ein eigenständiges Hobby. Doch auch als Hilfsmittel bei ganz anderen Interessengebieten sind Computer nützlich. Wir wollen euch hier Anwendungsmöglichkeiten für Sternengucker vorstellen — vielleicht euer neues Zweithobby?



Die nächste totale Mondfinsternis am 5. Mai 1985. oben: Berechnung links: Graphik des Verlaufs am Hamburger Himmel rechts: Mondfinsternis 1974

Natürlich kann man die Himmelskunde betreiben, ohne jemals mit einer einzigen mathematischen Formel in Berührung zu kommen. Ein tieferes Verständnis der Abläufe am Sternenhimmel erhält man jedoch nur dann, wenn man sich auch rechnerisch betätigt.

Besonders anschaulich und leicht verständlich für denjenigen, der sich mit der Himmelskunde bisher weniger befaßt hat, ist ein Teilgebiet, das etwas umständlich mit dem Begriff 'Phänomenologie' bezeichnet wird. In der Phänomenologie ist weniger die Ansicht des Sternenhimmels von Interesse, da ja die Sternbilder immer dieselbe Ansicht bieten und ihre Sichbarkeit nur einem jährlichen Wandel unterworfen ist. Erst die Bewegung anderer Gestirne wie Sonne, Mond und Planeten vor dem Hintergrund der Sterne führt zu eindrucksvollen Konstellationen. Dazu zählen das zeitweise enge Beieinanderstehen heller Planeten, das Vorbeimarschieren des Mondes an Planeten und Sternen, wobei diese häufig vom Mond bedeckt werden, und natürlich auch die Sonnen- und Mondfinsternisse.

Welchen Nutzen hat die Berechnung astronomischer Ereignisse überhaupt? Alle erwähnten Himmelsschauspiele sind, totale Sonnenfinsternisse einmal ausgenommen, wissenschaftlich heutzutage ohne großen Wert. Für den Hobby-Astronomen aber bilden diese oft mit bloßem Auge sichtbaren Geschehnisse einen ersten, faszinierenden Einblick in die Himmelskunde.

Voraussetzung für die Berechnung einzelner Gestirnskonstellationen ist die Bestimmung der Positionen der Himmelskörper in einem beliebigen Jahr, an einem beliebigen Tag und zu beliebiger Stunde, Minute und Sekunde. Zu diesem Zwecke gibt es Formelsammlungen, die eine Rechnung mit hoher Genauigkeit zulassen. Der vom Autor benutzte Algorithmus ermöglicht sogar eine Positionsbestimmung in einem enorm langen Geschichtsintervall, nämlich von 1000 v. Chr. bis 3000 n. Chr.. Innerhalb dieses Zeitraumes können Sonnen- und Mondfinsternisse, Sternbedeckungen durch den Mond, Auf- und Untergänge an beliebigen Orten der Erde usw. minutengenau (!) bestimmt werden. Hieraus ergeben sich faszinierende Möglichkeiten, Himmelsereignisse sowohl vorherzusagen als auch rückzudatieren.

Zur Bewältigung der zahlreichen Rechenmöglichkeiten wurden zwei Programme geschrieben. Das eine, genannt "PLANETARIUM", befaßt sich mit dem Lauf und den Konstellationen der Sonne sowie der mit dem bloßen Auge sichtbaren Planeten Merkur, Venus, Mars, Jupiter und Saturn. Das andere Programm heißt "LUNARIUM" und schließt neben den anderen Gestirnen die Bewegung des Mondes mit ein. Der Rechner, auf dem diese Programme in BASIC laufen, ist der Apple-Kupplung kompatible ITT 2020 mit 48 K im RAM.

Luna im alten Rom

Beide Programme, Variablen- und Programmspeicherplatz zusammengenommen, nutzen den verfügbaren RAM-Bereich (abzüglich DOS) weitgehend aus. Eine Aufteilung der verschiedenen Rechenroutinen in mehrere Programme wurde, zum Zwecke einer schnellen Zugriffsmöglichkeit zu unterschiedlichsten Berechnungsmöglichkeiten, vermieden. Da die Bestimmung von Gestirnpositionen innerhalb eines mehrere tausend Jahre langen Zeitraumes enorm viele ständig verfügbare Konstanten erfordert, wäre das bei einem Programmwechsel ständig wieder erforderliche Einlesen aller Daten allzu zeitraubend. Beide Programme, deren reiner Programmteil so bereits 16 K bzw. 22 K

am Homecomputer

lang ist, belegen für die ständig benötigten Daten schon einen Bereich von etwa 5 K.

Beim Apple gibt es aber bei derartig langen Programmen ein Problem, wenn wie im vorliegenden Falle gleichzeitig die hochauflösende Graphik benutzt werden soll. Da die Graphikseite mitten im üblicherweise vom Programm belegten Speicherbereich (zwischen $2000 und $4000) liegt, würde bei Benutzung des Graphikbefehls ein Großteil des Programms gelöscht werden. Dies wird umgangen, indem vom einmal fertiggestellten Programm mit einem Maschinenfile eine modifizierte Version erstellt wird. So wird der RAM-Bereich der Graphikseite in das Programm mit eingelagert, wobei dieser Bereich gleichzeitig gelöscht wird. Damit das BASIC-Programm trotzdem richtig läuft, wird der letzte Zeiger unterhalb von $2000 auf die erste Zeile oberhalb von $4000 gerichtet. Zum Laden eines so modifizierten Programms bedarf es eines weiteren kurzen Maschinenfiles, der zweckmäßigerweise in einem Aufruf-File untergebracht ist und für die korrekte Verbindung der Programmzeilen ab $4000 sorgt. Programme, in die eine Graphikseite eingelagert ist, werden um diese Seite (= ca. 8 K) länger. Der übrige RAM-Bereich ist nun vollständig für Programm und Variable nutzbar, was in einem 48 K-System etwa 28 K ausmacht.

Nur 3 Sekunden ...

Doch nun zurück zum Geschehen am Himmel. Um den Ort eines Gestirns anzugeben, muß eine Art Gradnetz vorhanden sein, in dem die jeweilige Position mit zwei Koordinaten festgelegt ist. Der gewünschte Berechnungszeitpunkt kann direkt als normales Datum und Uhrzeit eingegeben werden. Hieraus wird ein sogenannter Datumsfaktor ermittelt, aus dem sich wiederum die Bahndaten der Gestirne und ihre Position ergeben. Die ganze Rechnung dauert für jeden Planeten etwa 3 Sekunden.

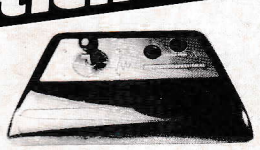
Welche Himmelsereignisse können nun berechnet werden? Zunächst alle Zusammenkünfte von Planeten. Auch für den Laien ist die gelegentliche Beobachtung zweier heller Lichtpunkte am Himmel, die manchmal sehr eng zusammenstehen können und innerhalb weniger Tage aneinander vorbeiziehen, ein eindrucksvolles Erlebnis, Der Rechner kann die jeweiligen Sichtbarkeitsbedingungen im Laufe eines beliebigen Jahres graphisch ermitteln. Hierzu werden die Positionen der Planeten schrittweise über einen bestimmten Zeitraum hinweg berechnet und im Koordinatensystem fortlaufend dargestellt. Stellt man an einer Stelle fest, daß sich zwei Planeten besonders nahe kommen, kann in einer anschließenden Rechnung das Rechen- sowie das Schrittintervall verkleinert werden und nach Eingabe eines Vergrößerungsfaktors und bestimmter Mittelpunktskoordinaten der betreffende Himmelsbereich fast beliebig herausvergrößert werden. Da der ITT 2020 in der hochauflösenden Graphik waagerecht 360 Bildpunkte hat, bietet sich in der Grunddarstellung des 360°-Himmels ein Maßstab von 1 Punkt pro Grad an. Mit den Formeln, die die Lage eines Graphikpunktes relativ zur Bildmitte bestimmen, kann durch die Veränderung des Vergrößerungsfaktors dann jeder andere Maßstab benutzt werden.

Die Frage nach den Sichtbarkeitszeiten bestimmter Planeten läßt sich hingegen am besten graphisch klären. Besonders eindrucksvoll ist dieses anhand der Planeten Merkur und Venus, die beide

Der Arkade Professional Super Joy Stick

compatibel mit
COMMODORE 64 & VC 20
ATARI 400, 800 & 600 XL
mit INTERFACE (Preis auf Anfrage)
SINCLAIR ZX 81 & SPECTRUM
DRAGON
TI 99/4A
COLECO

von EMAX COMPUTING

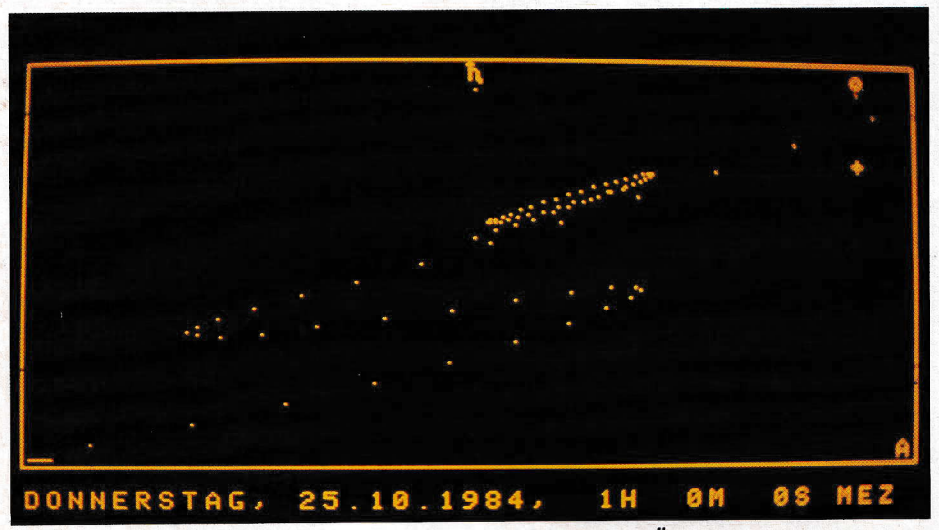


DM 139,-
incl. Mwst. + NN und Porto

mit entscheidenden Vorteilen
* für links- und rechtshänder geeignet
* 4 oder 8 Wege Steuerung
* 2 Schnellfeuerknöpfe
* Micro-Schalter
* standfest (36 × 20 cm)
* zerlegbar
Händleranfragen erwünscht.

Eckard Begerow
Electronic und Computer Zubehör-Vertrieb
Eichenstraße 11
8428 Rohr/Ndb.
Tel. 0 87 83/5 52

innerhalb der Erdbahn um die Sonne kreisen und daher immer nur in knappen Winkelabständen von der Sonne, also am morgendlichen oder abendlichen Dämmerungshimmel zu sehen sind. Hier sollte man als Berechnungszeitpunkt den Moment wählen, an dem die Sonne 5 Grad unter dem Horizont steht, d.h. kurz vor Sonnenaufgang oder kurz nach Sonnenuntergang. Das gewährleistet einen immer gleichhellen Himmelshintergrund, vor dem der Planet bei genügendem Horizontabstand deutlich sichtbar ist.



Schleifenbewegungen der Planeten Mars und Saturn im Äquatorsystem in diesem Jahr, Schrittintervall: 7 Tage.
Mars beschreibt die große Schleife in Bildmitte, Saturn die kleinere rechts oben. Die Planetensymbole sind als Shapes definiert.

Eine Möglichkeit, die beide Programme bieten, ist die Berechnung der Auf- und Untergangszeiten der Gestirne an einem beliebigen Ort der Erdoberfläche. Hierbei führen die Programme u.a. Näherungsrechnungen aus, die sogar die Lichtbrechung in der Erdatmosphäre berücksichtigen.

So läßt sich minutengenau der wahre Auf- und Untergangszeitpunkt bestimmen. Da bei dieser Rechnung die Gestirnposition mehrfach ermittelt werden muß, ergibt sich beispielsweise bei Planeten eine gesamte Rechenzeit von etwa 26 Sekunden.

Interessant ist auch die Beobachtung des Mondes, der ja Monat für Monat die Erde umrundet und dabei auch alle Planeten, helle Sterne und die Sonne überholt. Zusammenkünfte dieser Art gehören zu den eindrucksvollsten Himmelsereignissen überhaupt, zumal es auch, je nach Lage der Mondbahn, zu Bedeckungen der Gestirne kommt. Liegt bei Neumond die Sonne genau auf dem Weg des Mondes, was in der Regel zweimal im Jahr vorkommt, gibt es je nach Beobachtungsort eine partielle oder gar totale Sonnenfinsternis. Läuft der Mond in seiner Vollmondstellung einmal nicht über oder unter dem Erdschatten vorbei, sondern durch ihn hindurch, so erleben wir eine Mondfinsternis. Eine Rechenroutine, die alle Neumond- und Vollmondzeiten ermittelt und prüft, wie groß dann die Breitendifferenz zwischen Mond und Sonne bzw. Erdschatten ist, ist in der Lage, für ein beliebiges Jahr eine Liste der möglichen Finsternisse zu erstellen. Für jedes mögliche Datum kann jetzt mit einer weiteren Routine festgestellt werden, ob es an einem bestimmten Ort der Erde tatsächlich zu einer Finsternis kommt.

11. August 1999

Abweichungen der berechneten Finsterniszeiten zu den tatsächlichen liegen bei maximal 5 Minuten, was innerhalb einiger tausend Jahre sehr wenig ist. So wurden mit dem Programm schon Zeiten der nächsten totalen Sonnenfinsternis in Deutschland berechnet, die demnach am 11. August 1999 im süddeutschen Raum zu sehen sein wird. Auch die nächste totale Sonnenfinsternis in Norddeutschland konnte berechnet werden. Sie findet in den Mittagsstunden des 7. Oktober 2135 statt.

Der Vorübergang des Mondes an hellen Sternen und Planeten und deren Bedeckung ist auch sehr schön graphisch zu bestimmen. Die Bedeckung punktförmiger Fixsterne durch den atmosphärelosen Mond geschieht schlagartig. Zur Bedeckung kommt es in dem Moment, wo der Abstand zwischen Mond und Stern genau dem Mondradius entspricht. Bei Planetenbedeckungen hingegen benötigt der Mond eine Weile, um das kleine Planetenscheibchen ganz zu überlaufen. Analog zu einer totalen Finsternis werden dann bei Eintritt und Austritt des Mondes zwei Zeiten angegeben. Entsprechend kann, wie bei partiellen Finsternissen, der Mond einen Planeten auch nur tangieren, was zu einer teilweisen Abdeckung der Planetenoberfläche führt. Um dieses auf graphischem Wege in der Weise darzustellen, wie es in einem Fernrohr zu sehen wäre, muß eine stärkere Vergrößerung angewendet werden, um den Planeten als Scheibchen sichtbar zu machen. Bei dieser Darstellung erhält man gleichzeitig einen guten Eindruck vom Größenverhältnis des Mondes zum Planeten.

Die Möglichkeiten, die der Computer in der Berechnung der Geschehnisse am Himmel bietet, sind hiermit sicher noch nicht ausgeschöpft. Es konnte aber vielleicht gezeigt werden, daß für denjenigen, der sich eingehender mit Astronomie befassen will, der Computer ein ideales Hilfsmittel für die Vorbereitung auf bestimmte Beobachtungen und sogar für die Beschäftigung mit der Astronomiegeschichte durch Rückdatierung von Himmelsereignissen ist. Wenn die Beobachtungen eines Amateurs auch meist nicht von großem wissenschaftlichen Wert sind, so sind sie doch für ihn selbst interessant und können als Einstieg in ein faszinierendes Hobby dienen. Der Einsatz des Computers mit seinen Möglichkeiten der langfristigen Berechnung kann so zu einem tiefen Verständnis der ablaufenden Weltenmaschinerie führen.

Dr. Eberhard Riedel

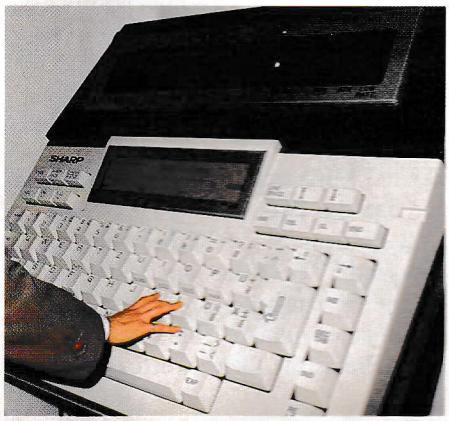
NEWS NEWS NEWS NEWS

COMPUTER Praxis auf der Hannover-Messe

Besucherrekord, verstopfte Straßen und Autobahnen und enorme Parkplatznot verdeutlichen den Erfolg der Hannover-Messe 1984. Das Team von COMPUTER Praxis hat sich durch die Besuchermassen gezwängt und einmal umgesehen was es alles so gab und sich über neue Geräte und Trends informiert. Für die nächsten Ausgaben gibt es viel aufzuarbeiten.

Eine kurze Information für unsere Leser aus dem offiziellen Messereport der Messeleitung: "Sehr ermutigend zu sehen war die Rolle, die die Jugendlichen auf dieser Hannover-Messe übernahmen. Von Technikfeindlichkeit oder Angst vor dem Computer war wenig zu spüren. Im CeBIT-Forum '84 'Berufe in der Büro- und Informationstechnik' regierten sachliches Interesse bis zur Neugier, intensives Informationsbedürfnis und schon beachtenswerter Sachverstand."

"Als Besuchermagneten erwiesen sich vor allem die vielfältigen Hard- und Software-Angebote bei den Mikrocomputern, die immer komfortabler werden, eine höhere Speicherfähigkeit aufweisen, mehrplatz- und multifunktionsfähig sind und Schnittstellen für Textverarbeitung, Personal-Computing, Graphik-Anwendung, Btx sowie Teletex aufweisen."



Besonders gefragt war die neue Ausgabe von COMPUTER Praxis auf der Messe.

Berichtigung des 17+4 Listings aus Heft 5/84

Durch einen Ausfall unseres Textsystems sind im Spiele Listing 17+4 im Atari-Teil zehn Zeilen nicht ausgedruckt worden. Beachtet werden muß auch, daß dieses Spiel nicht auf Computern mit weniger als 16K Speicherkapazität ohne größere Änderungen läuft.

```
1080 POSITION 31,2:? "WERT:";KWS;" "
1090 SPS=SPS+6:RETURN
1100 REM COMPUTER NIMMT KARTEN
1110 POSITION 1,15:? KC1$
1120 GOSUB 840:GOSUB 900
1130 IF KWC>=17 THEN RETURN
1140 GOTO 1120
1150 REM WAHLTAFEL
1160 FOR V=1 TO 50:NEXT V:ZEI=5
1170 POSITION 31,5:? "KARTE   "
```

Hinter den Schloßmauern findet im Ferienzentrum der Computerkurs statt.

Ferien mit Familie und Computer

Die Idee kommt — wie auch anders — aus den USA und hat auch hierzulande schnell Freunde gefunden:

Die Rede ist von Computercamps. Wißbegierige aller Altersklassen, vor allem aber Jugendliche, werden hier in den Ferien in individuellen Kursen spielerisch an den Computer herangeführt, oder können ihre bereits vorhandenen Kenntnisse vertiefen. Einer dieser Anbieter auf dem noch jungen Markt ist das Computercamp Ferienzentrum Schloß Dankern, im Emsland.

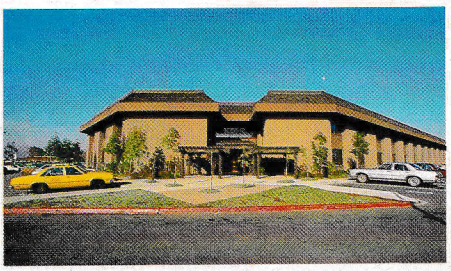
Laut Planungsteam legt man hier besonderen Wert auf sorgfältige Soft- und Hardware-Auswahl, und vor allem auf das "richtige" Freizeitangebot für die ganze Familie.

Besonders originell: Gewohnt wird nicht in Hotels oder Jugendherbergen, sondern etwas "lockerer" in Ferienhäusern (4—6 Personen).

Für weitere Informationen wendet man sich an: Planungsbüro Computercamp Ferienzentrum Schloß Dankern, Holztwiete 4 D, 2000 Hamburg 52, Telefon: (040) 82 79 42.

COMPUTER Praxis Heft 7

Ab 13. Juni beim Zeitschriftenhandel!



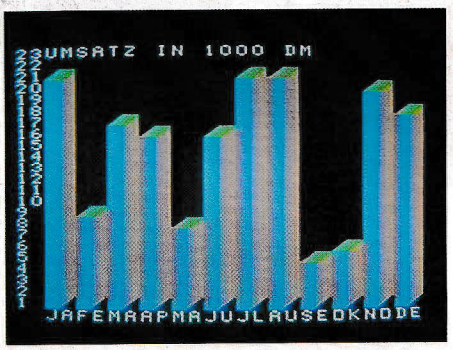
Das Atari-Hauptquartier im Silicon Valley beeindruckte uns sehr.

Neu von Sharp: der Pocket-Computer PC-1260. Das Wunderding ist voll BASIC-fähig.

Wie Wolkenkratzer sehen die Türme des Blockdiagramms aus.

Im neuen Heft findet ihr wieder jede Menge Listings, Hardware-Besprechungen und Testberichte über Profi-Software. Für den TI 99/4A haben wir unser erstes Listing eines Lesers aus der Schweiz erhalten. Der COMPUTER-Praxis-Fan bietet die Grundversion eines Spiels an, die von jedem nach seinen Wünschen erweitert werden kann. Kreativität ist hier besonders gefragt.

Weiter geht es auch in unserem BASIC-Kurs und mit der Maschinensprache des Z80.



Außerdem gibt es wieder jede Menge Tips und Tricks und vieles andere mehr.

Und: Wer einmal ein Programm veröffentlichen möchte, findet in COMPUTER Praxis die Gelegenheit dazu.

Inserentenverzeichnis

Atronic, Hamburg	67
BASF, Ludwigshafen	5
Eckard Begerow, Rohr	79
Commodore, Frankfurt	28, 29
Computercamp, Hamburg	13
Compy-Shop, Siershahn	37
ComZu, Riedstadt	57
Data Becker, Düsseldorf	24, 25, 33
Dynamics, Hamburg	73
Fujitsu, Frankfurt	4. US
Monika Glitsch, Leonberg	47
Haase, Essen	17
IWT-Verlag, Vaterstetten	59
Luther-Verlag, Sprendlingen	61
Marshall Cavendish, Hamburg	63
Mcgraw-Hill, Hamburg	31
Mikro-Händler, Mönchengladbach	83
Profisoft, Osnabrück	15
Sybex-Verlag, Düsseldorf	51, 53
te-wi-Verlag, München	43
Vecos-Warnecke, Hannover	21
Videostar, Hannover	11
Wella AG, Darmstadt	2. US
Wilke, Aachen	69
Winklers-Verlag, Darmstadt	45

Beilagenhinweis:
Einer Teilauflage unserer Zeitschrift liegt ein Prospekt der Firma Gruner + Jahr bei.

Impressum

COMPUTER Praxis
Redaktion
Paulstraße 3
2000 Hamburg 1
040 / 33 10 41

Verlag und Vertrieb
Marshall Cavendish International Ltd.
Niederlassung Deutschland
Heidenkampsweg 74
2000 Hamburg 1

Chefredaktion
Peter Aldick
(verantwortlich für den Inhalt)

Redaktion
Wiebke Krabbe, Holger Neuhaus

Ständige Mitarbeiter
Martin Albert, Gunnar Binder, Gerrit Kolb, Thomas Kregeloh, Jens-Peter Paulsen, Harald Uenzelmann, Carl Willke.

Layout
Uta Brandl

Produktion
Uta Brandl

Fotos
Rolf Seiffe

Titelillustration
Peter Fink

Anzeigen
Leitung: Christian Schlottau
(verantwortlich für Anzeigen)
Abwicklung: Loretta Arnold
Es gilt die Anzeigenpreisliste Nr. 1

Anschrift:
Heidenkampsweg 76 b, 2000 Hamburg 1, Telefon: 040 / 23 37 31 / 2

Satz
Atelier Schümann, Hamburg

Druck und Litho
Westermann, Braunschweig

Für unverlangt eingesandte Manuskripte und Disketten übernimmt der Verlag keine Haftung. Für die mit Namen der Autoren gekennzeichneten Artikel übernimmt die Redaktion lediglich die presserechtliche Verantwortung.
Für Fehler im Text, in Schaltbildern, Stücklisten usw., die zum Nichtfunktionieren oder Schadhaftwerden von Bauteilen oder Geräten führen, wird keine Haftung übernommen.
Mit Einsenden von Manuskripten, Fotos und Disketten übernimmt der Verlag das ausschließliche Recht der Veröffentlichung.

DAS SPANNENDE COMPUTERSPIEL VON MICRO HÄNDLER

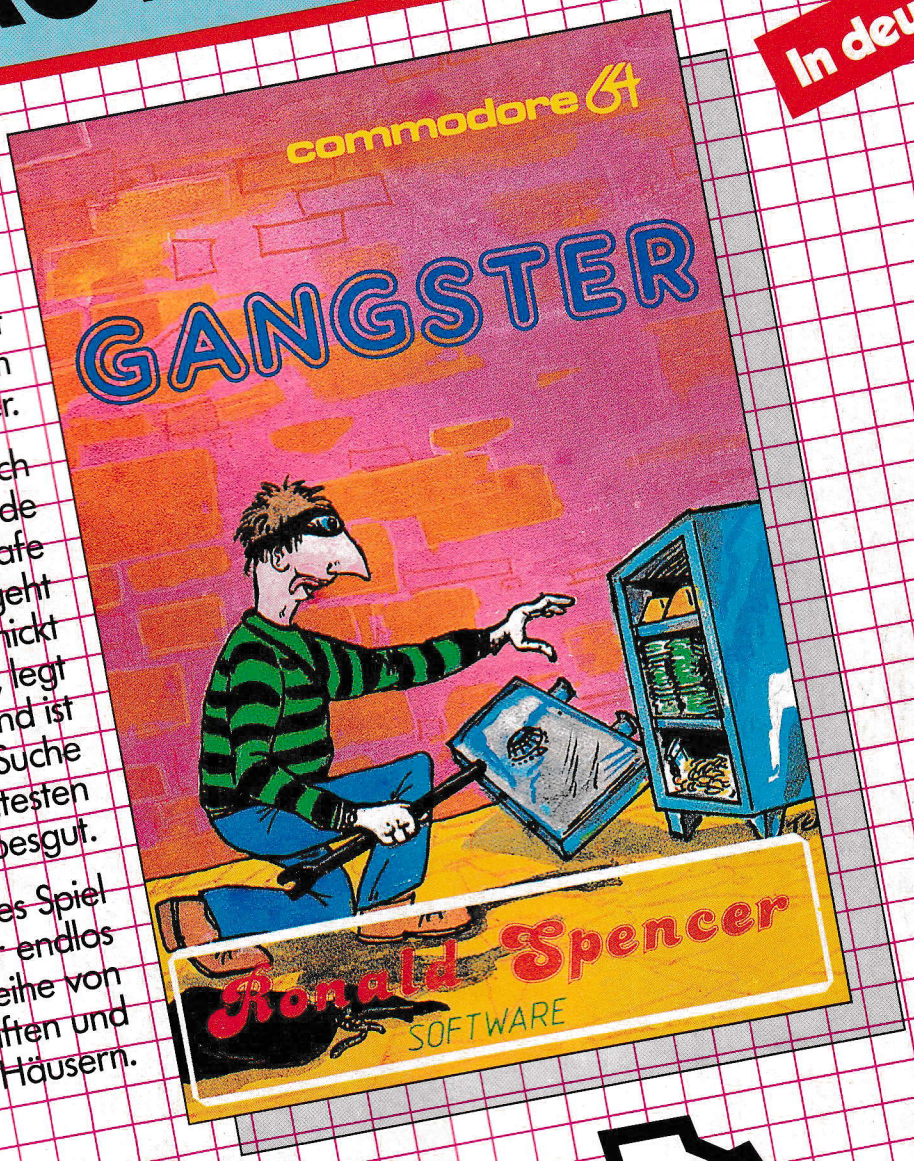
In deutsch!

GANGSTER

Das spannende Computerspiel mit Rodger the Dodger, dem Meistereinbrecher.

Bei seiner Tour durch zahlreiche Gebäude knackt er einen Safe nach dem anderen, geht Hindernissen geschickt aus dem Wege, legt Alarmsysteme still und ist immer auf der Suche nach dem entferntesten Diebesgut.

Ein faszinierendes Spiel mit einer endlos scheinenden Reihe von Banken, Geschäften und Häusern.

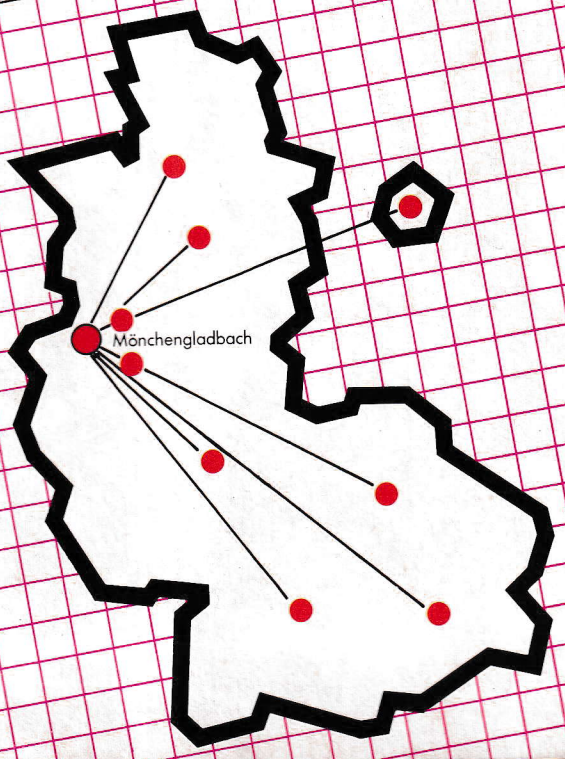


MICRO HÄNDLER

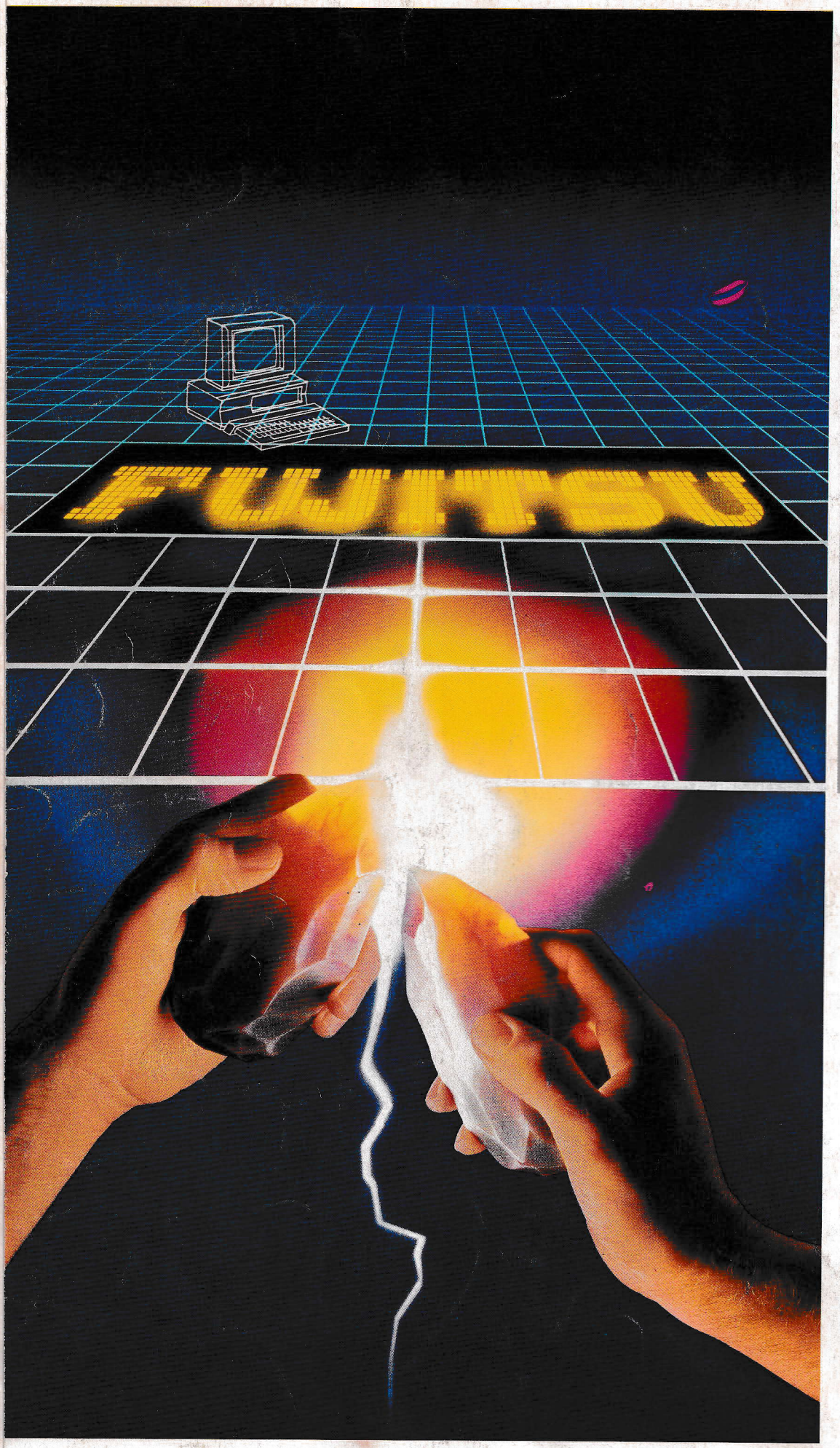
Bundesweite Lieferung an Händler

- **Große Auswahl**
- **Ständig Neuerscheinungen**
- **Sofortlieferung**

Robert-Koch-Str. 1
D-4050 Mönchengladbach 1
Tel. 0 21 61/65 13 88



Cr. Werbeteam GmbH · 4050 Mönchengladbach 1



Fujitsu, in Größe und Erfahrung führend in Japan.

富士通

In den Mikrocomputern von Fujitsu steckt die Erfahrung des größten japanischen Herstellers für Computer und Kommunikationstechnik.

Fujitsu Großcomputer sind die Nr. 1 in Japan. Einer der schnellsten universell einsetzbaren Rechner der Welt trägt unseren Namen. Und ohne Computersysteme von Fujitsu hätte es nicht schon 1964 den schnellsten Zug der Welt gegeben.

Fujitsu Telekommunikation ist führend in der Datenfernübertragung und Datenverteilung bis hin zur Glasfasertechnologie. In Japan ist mit Fujitsu die optische Übertragung von Nachrichten bereits Realität.

Fujitsu Bauelemente produziert hochintegrierte Mikrochips wie 256 kbyte RAM's sowie Tastaturen, Platten- und Diskettenlaufwerke. Die enorme Rechengeschwindigkeit und Speicherkapazität unserer Chips ermöglichen die hohe Leistungsfähigkeit moderner Computer.

Fujitsu Mikrocomputer vereinigen unsere Erfahrung in Computer- und Kommunikationstechnik und in der Herstellung wesentlicher Bauelemente. Fujitsu Mikrocomputer zeichnen sich durch hohe Zuverlässigkeit aus. Und das zu einem Preis, der in dieser Klasse neue Maßstäbe setzt.

Fujitsu - ein Funke springt über.

FUJITSU

Fujitsu Mikroelektronik GmbH,
Lyoner Straße 44-48,
D-6000 Frankfurt 71,
Telefon 0611/6632150